

• РАДИО И СВЯЗЬ •

СПРАВОЧНИК

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

КНИГА

I

СИСТЕМЫ
ОБЩЕНИЯ
И ЭКСПЕРТНЫЕ
СИСТЕМЫ

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ



СПРАВОЧНИК

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

В ТРЕХ КНИГАХ

1

**СИСТЕМЫ ОБЩЕНИЯ
И ЭКСПЕРТНЫЕ СИСТЕМЫ**

2

МОДЕЛИ И МЕТОДЫ

3

**ПРОГРАММНЫЕ
И АППАРАТНЫЕ СРЕДСТВА**

СПРАВОЧНИК

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

КНИГА

1

СИСТЕМЫ ОБЩЕНИЯ И ЭКСПЕРТНЫЕ СИСТЕМЫ

**ПОД РЕДАКЦИЕЙ
ПРОФЕССОРА
Э. В. ПОПОВА**



**МОСКВА „РАДИО И СВЯЗЬ”
1990**

ББК 32.81
И 86
УДК 007 : 159.955 (035)

Рецензенты: д-р техн. наук *Б. А. Шукин*, канд. техн. наук
О. С. Констанденко

Редакция литературы по информатике и вычислительной технике

Искусственный интеллект: В 3 кн. Кн. 1. Системы об-
И86 щения и экспертные системы: Справочник /Под ред.
Э. В. Попова. — М.: Радио и связь, 1990. — 464 с.:
ил.

ISBN 5-256-00365-8(кн. 1).

Приводится классификация интеллектуальных систем по их структуре и решаемым задачам. Описываются особенности систем общения и экспертных систем, получивших наиболее широкое распространение на практике. Рассматривается состояние развития систем общения (естественно-языковых, систем речевого общения, обработки визуальной информации, машинного перевода), экспертных систем и инструментальных систем для их создания.

Для специалистов в области управления, информационных систем и вычислительной техники, использующих методы искусственного интеллекта.

И 1402070000-048
046(01)-90 27-90

ББК 32.81

ISBN 5-256-00365-8(кн. 1)
ISBN 5-256-00756-4

© Коллектив авторов, 1990

Предисловие

Искусственный интеллект — одно из самых молодых научных направлений. Прошло немногим более четверти века с момента начала работ в этой области. Но сегодняшний мир и научно-технический прогресс немыслимы без тех результатов, которые дали человечеству и обещают дать в будущем исследования в области искусственного интеллекта.

Причин для столь исключительного положения работ в области искусственного интеллекта несколько. Наиболее важными из них являются следующие:

1. Развитие вычислительной техники вступило в новую фазу — создание ЭВМ пятого и последующих поколений. Отличительной чертой новых ЭВМ является их максимальная приближенность к пользователю, освобождение пользователя от программирования решения задач. Функции программиста передаются ЭВМ, сложность общения с ней не должна превосходить сложности общения с современными бытовыми системами. Для этого необходимо поднять «интеллектуальный» уровень ЭВМ, сделав ее способной к выполнению творческого профессионального труда программиста. В ЭВМ должна быть заложена большая сумма знаний о способах решения задач, специальные процедуры автоматического синтеза программ, а также средства общения с пользователем, максимально приближающие это общение к общению людей.

2. XXI век нередко называют веком новых информационных технологий. В связи с массовым внедрением ЭВМ во все сферы человеческой жизни, развитием локальных, глобальных, национальных и межгосударственных сетей передачи и обработки данных, быстрым ростом хранилищ информации, распределенных в различных местах земного шара и доступных для любой ЭВМ, включенной в соответствующую сеть, станет возможным переход к безбумажной технологии обработки информации. А это повлечет за собой изменение стиля человеческого общения в самых разнообразных деловых и бытовых взаимоотношениях.

3. Технология производства в промышленности и сельском хозяйстве меняет свою основную парадигму. До сих пор она ориентировалась на человека как основное звено в трудовом процессе. Роботизация производства заставляет по-новому оценить организацию трудового процесса. В роботизированных производствах нет необходимости в создании «человеческих условий» для производителей, в выполнении требований к агрегатам, климатическим условиям и т. п., которые определяются эргономикой и физиологией человека.

4. В проектировании новых образцов изделий и в научных исследованиях интеллектуальные системы должны сыграть в ближайшие годы революционизирующую роль. Они являются тем инструментом, без которого станет невозможным проектирование сверхсложных для человека изделий и который позволит ученым в приемлемые сроки решать те научные проблемы, на решение которых не хватает сейчас жизни нескольких поколений.

Можно указать еще ряд причин (повышение эффективности обучения, создание роботов-исследователей, работающих в средах, непригодных для человека и т. п.), но и сказанного достаточно, чтобы почувствовать глобальную значимость того направления, которое называется «искусственный интеллект».

Вряд ли сейчас возможно какое-либо единое определение, полностью описывающее эту научную область. Среди многих точек зрения на нее сегодня доминируют три. Согласно первой исследования в области искусственного интеллекта являются фундаментальными исследованиями, в рамках которых разрабатываются модели и методы решения задач, традиционно считавшихся интеллектуальными и не поддававшихся ранее формализации и автоматизации. Согласно второй точке зрения новое направление связано с новыми идеями решения задач на ЭВМ, с разработкой принципиально иной технологии программирования, с переходом к архитектуре ЭВМ, отвергающей классическую архитектуру, которая восходит еще к первым ЭВМ. Наконец, третья точка зрения, по-видимому, наиболее прагматическая, состоит в том, что в результате работ в области искусственного интеллекта рождается множество прикладных систем, решающих задачи, для которых ранее создаваемые системы были непригодны.

Конечно, все эти три точки зрения взаимно связаны. В области искусственного интеллекта развиваются фундаментальные исследования, новая технология программирования, новая архитектура технических средств, и все это используется для создания прикладных систем, предназначенных для работы в самых разнообразных областях.

Настоящий справочник по искусственному интеллекту отражает указанные точки зрения. Три его книги как раз и соответствуют им. В первой книге «Искусственный интеллект. Системы общения и экспертные системы» описаны принципы построения естественно-языковых интерфейсов и средств графического общения, приведены общие сведения о технологии построения экспертных систем и примеры наиболее известных интеллектуальных систем, разработанных в СССР. Этот том предназначен не только для специалистов, работающих в области искусственного интеллекта, но и для широкого круга лиц, интересующихся использованием интеллектуальных систем в тех областях, в которых они работают.

Во второй книге «Искусственный интеллект. Модели и методы» помещены материалы, касающиеся извлечения знаний, формализации их и представления в интеллектуальных системах, способов манипулирования знаниями при поиске решений, проблем общения, работы с изображениями и распознавания образов, а также проблем обучения и планирования целесообразного поведения. Эта книга может служить и учебным пособием для студентов, специализирующихся в области искусственного интеллекта, и аспирантов, работающих в этой области. Она будет интересна и специалистам, поскольку представляет собой сжатую сводку результатов теории искусственного интеллекта, накопленных к настоящему времени.

Третья книга «Искусственный интеллект. Программные и аппаратные средства» содержит описание базовых средств, которые используются при программировании задач в интеллектуальных системах, языков представления знаний и манипулирования ими, инструментальных систем, используемых при создании проблемно-ориентированных интеллектуальных систем, а также спецпроцессоров, архитектура которых ориентирована на реализацию тех функций, которые характерны для интеллектуальных систем. Эта книга рассчитана в первую очередь на специалистов, разрабатывающих интеллектуальные системы и инструментарий для них, а также на студентов и аспирантов, обучающихся по специализациям, связанным с искусственным интеллектом и созданием ЭВМ новых поколений.

Справочников по интеллектуальным системам пока нет ни в СССР, ни в других странах. Поэтому подготовка материалов для него потребовала большого труда многих специалистов нашей страны. Окончательную подготовку справочника осуществляли специальная редколлегия и рабочая группа, назначенные Научным советом по проблеме «Искусственный интеллект» Отделения информатики, вычислительной техники и автоматизации АН СССР. В состав редколлегии входили: академик Г. С. Поспелов (председатель), д-р техн. наук проф. Д. А. Поспелов (зам. председателя), канд. техн. наук В. Ф. Хорошевский

(ученый секретарь), д-р техн. наук, проф. Э. В. Попов, канд. техн. наук В. Н. Захаров. В состав рабочей группы входили: канд. физ.-мат. наук А. Н. Аверкин, канд. техн. наук А. М. Волков, канд. техн. наук М. Г. Гаазе-Рапопорт, В. Н. Дембовская, канд. физ.-мат. наук С. М. Ефимова, канд. техн. наук Б. С. Кирсанов, канд. физ.-мат. наук Л. В. Литвинцева, канд. техн. наук А. Б. Преображенский, Н. В. Руссова, Н. В. Чудова.

Все замечания по содержанию справочника просим направлять в Научный совет по проблеме «Искусственный интеллект». Отделения информатики, вычислительной техники и автоматизации АН СССР по адресу: 117333 Москва ул. Вавилова, 40.

Предисловие редактора

В 80-е годы на основе исследований в области искусственного интеллекта (ИИ) сформировалась новая отрасль индустрии — производство интеллектуальных систем. Интеллектуальные системы предназначены для выполнения на ЭВМ таких практических задач, которые называются интеллектуальными, если они выполняются людьми. Примерами интеллектуальных являются задачи понимания и синтеза текстов на естественном языке, понимания и синтеза речи, анализа, обработки и синтеза изображений, перевода с одного естественного языка на другой, принятия решений в условиях изменяющегося окружения и т. п.

Все существующие интеллектуальные системы можно разбить на два класса: общего назначения и специализированные. К интеллектуальным системам общего назначения отнесены те, которые не только исполняют заданные процедуры, но на основе метапроцедур поиска генерируют и исполняют процедуры решения новых конкретных задач. Технология использования таких систем состоит в следующем. Пользователь (эксперт) формирует знания (данные и правила), описывающие выбранное приложение. Затем на основании этих знаний, заданной цели и исходных данных метапроцедуры системы генерируют и исполняют процедуру решения конкретной задачи. Данную технологию называют технологией систем, основанных на знании, или технологией инженерии знаний. Она позволяет специалисту, не знающему программирования, разрабатывать гибкие прикладные системы.

В настоящее время, пожалуй, единственным типом интеллектуальных систем, которые могут быть отнесены к классу систем общего назначения, являются оболочки экспертных систем.

К специализированным интеллектуальным системам отнесены те, которые выполняют решение фиксированного набора задач, predeterminedного при проектировании системы. Для использования таких систем требуется наполнить их данными, соответствующими выбранному приложению. До недавнего времени при разработке специализированных систем использовалась технология традиционного (процедурного) программирования, что позволяло обеспечить их высокую эффективность. Однако эта технология существенно ограничивала способность систем к изменению их поведения при изменяющемся окружении, что крайне важно для решения многих интеллектуальных задач. В связи с этим в последнее время с целью устранения этого недостатка специализированные интеллектуальные системы (системы речевого общения, системы обработки изображений) стали разрабатывать, используя технологию инженерии знаний, в виде экспертных систем.

Данная книга посвящена описанию современного состояния разработки систем общения и экспертных систем, т. е. той разновидности интеллектуальных систем, которые получили наиболее широкое распространение на практике. К системам общения отнесены системы обработки текстов естественного языка (ЕЯ-системы), системы речевого общения, системы обработки визуальной информации и системы машинного перевода. К экспертным системам отнесены собственно экспертные системы и оболочки экспертных систем.

При рассмотрении каждого из перечисленных выше типов систем использован следующий порядок изложения: 1) особенности, характеризующие системы данного типа; 2) обзор систем; 3) описание отечественных систем данного типа. Детальное описание именно отечественных, а не зарубежных систем предпринято с той целью, чтобы читатели могли не только ознакомиться с принципами и методами построения интеллектуальных систем, но и получить для использования заинтересовавшую их систему. При отборе отечественных систем предпочтение отдавалось тем, которые находятся на промышленной или коммерческой стадии. При этом под промышленной понимается такая стадия существования системы, которая характеризуется как наличием документации, выполненной по требованиям ЕСКД или ЕСПД, так и проведением соответствующих испытаний, под коммерческой — стадия, следующая за промышленной и характеризующая наличием организации (типа СНПО «Алгоритм»), осуществляющей поддержку и распространение системы по заявкам заинтересованных организаций.

Книга состоит из двух частей. Первая часть, охватывающая гл. 1—12, содержит описание и обзор интеллектуальных систем общения: систем обработки текстов естественного языка (гл. 1—3), систем речевого общения (гл. 4—6), систем обработки изображений (гл. 7—9) и систем машинного перевода (гл. 10—12). Необходимо отметить, что подавляющее большинство существующих систем общения, как отечественных, так и зарубежных, построены как специализированные интеллектуальные системы с использованием технологии процедурного программирования.

Вторая часть, охватывающая гл. 13—18, посвящена рассмотрению на примере экспертных систем интеллектуальных систем общего назначения. Здесь рассмотрены вопросы проектирования экспертных систем (гл. 13), приведен обзор экспертных систем и инструментальных средств, предназначенных для их создания (гл. 14). Завершается вторая часть рассмотрением отечественных «оболочек» (гл. 15 и 16), находящихся на промышленной или коммерческой стадии, и экспертных систем, созданных на их основе (гл. 17 и 18).

Необходимо подчеркнуть, что данная книга не охватывает всех существующих типов интеллектуальных систем. Так, за ее рамками остались системы, используемые в САПР и САНИ, системы ИИ для решения задач в области планирования и проектирования. Предполагается, что перечисленные системы ИИ будут описаны в последующих томах справочника.

Книга ориентирована на широкий круг пользователей ЭВМ, которые, ознакомившись с ней, могут получить представление о возможностях существующих интеллектуальных систем и о том, насколько эти системы могут удовлетворить их потребности. Она будет полезна и разработчикам автоматизированных систем, не использовавшим ранее методы искусственного интеллекта в своей работе.

Глава 1.

Особенности реализации ЕЯ-систем

Э. В. Попов, А. Б. Преображенский

1.1. Назначение и область применения ЕЯ-систем

В конце 60-х годов в исследованиях по ИИ сформировалось самостоятельное направление, получившее название «обработка естественного языка» (Natural Language Processing). Задачей данного направления является исследование методов и разработка систем, обеспечивающих реализацию процесса общения с ЭВМ на естественном языке (систем ЕЯ-общения, или ЕЯ-систем). Следует отметить, что проблематика коммуникативного взаимодействия, и в частности ЕЯ-общения, находится в центре внимания многих наук, например лингвистики, психологии, логики и философии. Однако все они исследуют лишь отдельные аспекты процесса общения. В отличие от них ИИ как прикладная дисциплина вынуждена моделировать в рамках ЕЯ-систем все или по крайней мере основные аспекты ЕЯ-общения, правда, может быть не на столь глубоком уровне.

Проблема взаимодействия человека с ЭВМ существует с момента появления вычислительной техники. На начальном этапе непосредственное взаимодействие с ЭВМ осуществляли только программисты, а специалисты других областей — потребители результатов, полученных на ЭВМ, выступали в роли косвенных конечных пользователей, т. е. общались с ЭВМ через программистов. По мере расширения сферы использования ЭВМ и увеличения масштабов их применения конечные пользователи стали вовлекаться в процесс непосредственного взаимодействия с ЭВМ, что привело к появлению массовой категории пользователей — прямых конечных пользователей, работающих в диалоговом режиме [Martin, 1981]. К пользователям этой категории относятся специалисты в различных проблемных областях, которые решают свои профессиональные задачи, непосредственно используя ЭВМ, т. е. прикладные программы и пакеты (прикладные системы), входящие в программное обеспечение ЭВМ. Как правило, эти пользователи не обладают знаниями в области электронной обработки данных и не умеют программировать. Поэтому часто их называют неподготовленными конечными пользователями. В дальнейшем термины «пользователь» и «конечный пользователь» будут использоваться в смысле «неподготовленный конечный пользователь».

Сложность создания средств общения, предназначенных для конечных пользователей, обусловлена в значительной степени отсутствием единой теории языкового общения, охватывающей все аспекты взаимодействия коммуникантов. Поэтому при разработке средств общения конечных пользователей на процесс взаимодействия часто налагаются различные «спонтанные» ограничения, последствия которых не до конца осознаются разработчиками. Эти ог-

раничения приводят к тому, что многие человеко-машинные системы, на разработку которых тратятся огромные средства (например, большие автоматизированные системы обработки данных типа АСУ), не удовлетворяют требованиям конечных пользователей.

Прежде чем приступить к рассмотрению ЕЯ-систем, остановимся на определении некоторых исходных понятий, которые будут широко использоваться в книге.

Общение — коммуникативное взаимодействие, *диалог* — процесс достижения его участниками определенных согласованных целей путем обмена связанными высказываниями, выраженными в языке, о некотором реальном или гипотетическом мире (проблемной области) [Попов, 1982]. Говоря об общении человека с ЭВМ, обычно считают (см. [Виноград, 1983; Новое в лингвистике, 1983; Shank et al., 1979]), что цель ЭВМ состоит в том, чтобы способствовать достижению целей пользователя, которые определяются его информационными потребностями. Поэтому применительно к *диалогу между пользователем и ЭВМ* под общением понимают процесс обмена взаимосвязанными высказываниями, выраженными в языке, направленный на достижение целей пользователя, т. е. на удовлетворение информационных потребностей пользователя (ИПП).

В общем случае процесс общения не может быть сведен к обмену изолированными парами высказываний вопрос-ответ. Высказывания участников общения образуют связный текст — *дискурс*, имеющий, как правило, достаточно сложную структуру. Связность дискурса обеспечивается как лингвистическими (родовидовыми, анафорическими, модальными, стилистическими согласованиями, согласованиями presupпозиций и т. п.), так и экстралингвистическими (ситуативными) средствами, т. е. с помощью временных, причинно-следственных и других связей, существующих в проблемной области [Звягинцев, 1976]. Следует подчеркнуть, что разговорный ЕЯ гораздо более компактен, чем литературный, «письменный» язык, так как при общении широко используются разнообразные умолчания (эллипсисы, анафоры presupпозиции и др.), восстанавливаемые (раскрываемые) участниками исходя из текущих целей диалога.

Если участники достигли цели, поставленные в начале общения, то говорят, что общение завершилось *успехом* (глобальным успехом), в противном случае — *неудачей* (глобальной неудачей). В процессе общения могут возникать различные *локальные неудачи*, вызванные, например, неправильностью (нарушением грамматических норм) высказываний участников, непониманием друг друга из-за различных представлений о теме диалога или о проблемной области и языке общения и т. п. Большая часть локальных неудач не приводит к глобальной неудаче; они преодолеваются участниками общения гибкой (в ходе диалога) корректировкой текущих целей.

Цели, преследуемые участниками общения, определяют структуру диалога, которая может рассматриваться по крайней мере на трех уровнях: глобальном, тематическом и локальном [Grosz, 1977]. На *глобальном уровне* (общая структура) определяются общие свойства решаемых пользователями задач. На *тематическом уровне* структура диалога зависит от конкретных особенностей решаемой задачи — от алгоритма ее решения (разбиения задач на подзадачи) и распределения ролей (активная или пассивная роль) между участниками общения при решении отдельных подзадач. На *локальном уровне* рассматриваются отдельные шаги диалога, образуемые взаимосвязанными высказываниями его участников. *Шаг диалога* трактуется как пара действие-реакция, где высказывание активного (т. е. владеющего инициативой) участника соответствует действию, а пассивного — реакции [Деннинг и др., 1984]. Основными параметрами структуры диалога на этом уровне являются: инициатор шага и вид инициирования (вид действия); способ влияния действия на реакцию; способ спецификации задачи (подзадачи), решаемой на данном шаге. Действие и реакция, образующие шаг диалога, могут в общем случае не соответствовать соседним (во временной последовательности) высказываниям участников. Соответствие нарушается при перехватах инициативы.

Перехват инициативы возникает в тех случаях, когда пассивный участник вместо цели (подцели), предложенной активным участником, выбирает иные цели (подцели), в частности подцели, предусматривающие преодоление локальных неудач. Например, вместо ответа на вопрос (что соответствовало бы стандартной реакции) второй участник может задать встречный вопрос (т. е. совершить действие и тем самым взять на себя активную роль) и лишь после получения ответа на него ответить на первоначально заданный вопрос (и тем самым вернуть инициативу). Таким образом, перехват инициативы как бы разрывает первоначально инициированный шаг диалога и открывает поддиалог — происходит смена цели (темы) диалога, в котором инициативой владеет ранее пассивный участник.

Переходя к рассмотрению человеко-машинного общения, подчеркнем, что, согласно современным представлениям (см. [Дракин и др., 1988; Martin, 1981]), взаимодействие конечных пользователей с ЭВМ происходит на всех стадиях существования человеко-машинной системы, т. е. на стадиях использования, разработки и развития приложений. Традиционные средства общения, которые вплоть до настоящего времени широко применяются на практике, ориентированы, как правило, либо только на использование заранее разработанных и неизменяемых приложений (см., например, система по продаже авиационных или железнодорожных билетов), либо на использование, разработку и развитие приложений (см., например, язык ДПС системы ДИСОД [Система ДИСОД, 1987, § 7.1]). В первом случае процесс взаимодействия сводится к трем этапам: определение параметров работы системы (вход в систему); определение решаемой задачи и исходных данных; получение результатов решения задачи — и принципиально не может удовлетворить пользователей с изменяющейся информационной потребностью, не знающих способа представления и использования в системе информации, которой обмениваются участники общения. Во втором случае взаимодействие осуществляется с помощью процедурного языка программирования, что не удовлетворяет большинство конечных пользователей, обычно не умеющих (и не желающих) программировать.

Низкая эффективность, а часто неприемлемость традиционных средств общения в большинстве случаев вызвана тем, что в них не учитываются важнейшие особенности процесса общения, направленного на удовлетворение реальных ИПП. Эти особенности независимо от специфики решаемых пользователями задач сводятся к следующим.

1. Изменяемость. ИПП не может быть заранее четко определена в спецификациях на разработку системы общения, напротив, ИПП неизбежно изменяется в ходе разработки и эксплуатации системы.

2. Несовпадение взглядов на мир. Представления, имеющиеся у пользователя и системы о языке общения и проблемной области, относительно которой ведется общение, могут не совпадать. Исходя из этого, процесс общения должен предусматривать разъяснение смысла неизвестных терминов, обнаружение и устранение несовпадающих представлений, а также предупреждение ошибочных толкований, т. е. установление общих точек зрения на обсуждаемые в процессе общения сущности.

3. Связность общения. Процесс общения не может быть ограничен обменом изолированными парами «вопрос-ответ», так как в большинстве реальных случаев ИПП не может быть выражена в виде одного вопроса (предложения). Часто требуется определить ситуацию, в которой возникла ИПП, т. е. предположить запросу на решение некоторой задачи контекст, в котором эту задачу необходимо решать. Кроме того, процесс удовлетворения ИПП — решение некоторой задачи, в большинстве реальных приложений требует взаимодействия, основанного на смешанной инициативе участников. Поэтому процесс общения должен иметь сложную, разветвленную структуру и состоять из обмена связанными высказываниями.

4. «Неправильность» высказываний пользователя. Для выражения ИПП пользователь может применять как «правильные» предложения, т. е. такие, которые будут однозначно поняты и верно обработаны системой, так и «непра-

вильные». Неправильности могут быть вызваны, во-первых, тем, что пользователь обычно не в состоянии учесть все ограничения системы общения в части ее возможностей и знаний, во-вторых, использованием умолчаний, характерных для естественного общения и допускающих неоднозначное толкование высказываний, и, в-третьих, отклонением предложений от грамматической нормы.

Недостатки традиционных средств общения потребовали создания средств нового поколения, которые должны быть способны настраиваться на ИПП и адаптироваться к их изменению, представлять и объяснять свою точку зрения на проблемную область, а также учитывать точку зрения пользователя, поддерживать связный диалог и уметь обрабатывать «неправильные» высказывания. Разработка этих средств ведется в настоящее время по двум основным направлениям. Первое направление, развиваемое преимущественно специалистами по системам обработки данных, заключается в повышении уровня и увеличении непроедурности формализованных языков общения. Типичными представителями таких языков являются, например, APL, NOMAD, MAPPER, QBE [Martin, 1981]. Второе направление развивается в рамках ИИ и предполагает использование конечными пользователями для взаимодействия с ЭВМ естественного языка, семантически и прагматически ограниченного проблемной областью, относительно которой ведется общение. Рассматриваемые далее ЕЯ-системы разработаны в рамках второго направления.

1.2. Обобщенная схема ЕЯ-системы

Традиционные средства общения не позволяют обеспечить взаимодействие конечных пользователей с ЭВМ. Чтобы быть полноправным участником общения, ЕЯ-система должна выполнять некоторые обязательные функции. К этим функциям относятся, по крайней мере, следующие: *ведение диалога* — определение его структуры и той роли, которую система и пользователь выполняют на текущем шаге диалога; *понимание* — преобразование поступающих от пользователя высказываний на ЕЯ в высказывания на языке внутреннего представления; *обработка высказываний* — формирование или определение заданий на решение задач или подзадач на данном шаге диалога; *генерация* — формирование выходных высказываний на ЕЯ. Приведенные функции имеют обобщенный характер. Поэтому необходимо подчеркнуть, что при реализации конкретных ЕЯ-систем суть этих функций может в значительной степени варьироваться.

В соответствии с выделенными функциями обобщенная схема ЕЯ-системы (рис. 1.1) может быть представлена в виде трех компонентов: диалоговый; компонент понимания высказываний; компонент генерации высказываний.

К основным задачам диалогового компонента относятся: ведение диалога; формирование или обработка перехватов инициативы. Ведение диалога состоит в том, чтобы обеспечивать целесообразные (т. е. способствующие достижению конечных целей пользователя) действия системы на текущем шаге диалога. В связи с тем, что возможности существующих ЕЯ-систем не позволяют им самостоятельно формировать целесообразное поведение, в систему обычно вводится информация, определяющая общую и тематическую структуры диалога. По структуре и текущему состоянию диалога диалоговый компонент формирует (если инициатива принадлежит системе) или определяет (если инициатива принадлежит пользователю) задание, выполняемое системой на текущем шаге (например, генерация вопроса, понимание ответа и его обработка, генерация утверждения и т. п.).

Ведение диалога выполняется по одной из двух схем: диалог ведет пользователь; диалог ведет система. В первом случае инициатива в основном (за исключением сообщений об ошибках) находится у пользователя, а система только реагирует на его требования, определяя по виду требования тип задания. Разбиение задачи на подзадачи и распределение ролей осуществляет пользователь, т. е. для системы весь диалог сводится к выработке реакций на те-

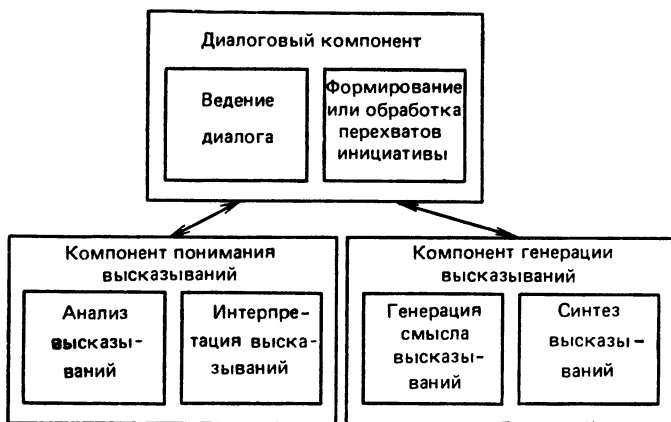


Рис. 1.1. Обобщенная схема ЕЯ-системы

кущие высказывания пользователя. Можно сказать, что в этом варианте работы функции диалогового компонента практически вырождены. Во втором случае инициатива в основном принадлежит системе. Система ведет диалог в соответствии с имеющимися у нее представлениями о структуре диалога (т. е. о разбиении задач на подзадачи и о том, кто из участников, когда и какую подзадачу решает) и о способе обмена высказываниями.

Если роли участников неизменны, однозначны и предопределены заранее, то структуру диалога называют *жесткой*. В простейшем случае такая структура диалога сводится к двум взаимосвязанным высказываниям участников (вопрос-ответ) с указанием участника, владеющего инициативой. Развитием жесткой структуры является *альтернативная структура*, которая задает множество возможных (но заранее предписанных) направлений течения диалога. Выбор одного из возможных направлений осуществляет пассивный участник. Если роли участников общения распределяются в ходе общения, то структуру диалога называют *гибкой*. Гибкие структуры подразделяются по степени свободы выбора момента перехвата (предопределенные моменты, произвольные моменты) и способа перехвата (предопределенный способ перехвата, произвольный способ).

Вторая задача диалогового компонента вызвана тем, что реакции одного участника могут не соответствовать ожиданиям другого. В зависимости от того, кто осуществляет перехват инициативы, система либо формирует перехват, либо обрабатывает его. Формирование происходит в тех случаях, когда система определяет, что текущая ситуация не соответствует ситуации, предусмотренной структурой диалога. Если же перехват инициативы осуществляет пользователь, то задача системы — обработать его, т. е. распознать наличие перехвата инициативы, определить новую тему (цель), на которую перешел пользователь, и перейти на структуру диалога, соответствующую новой теме.

Компонент понимания высказываний предназначен для выделения смысла входного высказывания и выражения этого смысла на внутреннем языке системы. Под *смыслом высказывания* обычно понимается вся та семантико-прагматическая информация, которую пользователь хотел передать системе. Внутреннее представление смысла должно содержать, по крайней мере, следующую информацию: сущности проблемной области, вовлекаемые в зону рассмотрения данным высказыванием; свойства и отношения, приписанные этим сущностям; коммуникативные намерения говорящего, выраженные в данном высказывании. Выявление смысла высказывания в общем случае требует его рассмотрения в контексте всего диалога.

Таблица 1.1.

Общая характеристика анализаторов ЕЯ-систем

Параметр		Возможные значения	
Типы анализируемых предложений		Повествовательные, вопросительные, отрицательные, полные, неполные, простые, сложные, распространенные, нераспространенные и др.	
Выделяемые описания сущностей	Понятия		Конкретные (индивидуальные), абстрактные, метапонятия
	Отношения	Предикаты	Вспомогательные, предикаты-состояния, предикаты-действия, функциональные и др.
		Кванторы	Отсутствие кванторов, кванторы существования всеобщности, отрицания
		Модальности	Отсутствие модальностей, объективная и субъективная
	Пресуппозиции		Отсутствие пресуппозиций, экзистенциальные, семантические, прагматические
Глубина проникновения в смысл		Множество ключевых слов, имя события и описания участников события (их роли и характеристики), сценарий с отсылками к связанным под-сценариям, пространственно-временное или причинное представление ситуации	
Используемые средства		Морфологический, синтаксический, семантический, прагматический анализ	

Традиционно задача понимания высказываний подразделяется на два этапа: анализ и интерпретацию. На этапе *анализа* выделяются описания сущностей; упомянутых во входном высказывании, выявляются свойства этих сущностей и отношения между ними. Анализ обычно выполняется отдельным блоком — анализатором, служащим ядром компонента понимания. Анализаторы, разрабатываемые для ЕЯ-систем, различаются по ряду параметров. Основные из них: тип анализируемых предложений; выделяемые описания сущностей; глубина проникновения в смысл; используемые для анализа средства. Возможные значения этих параметров приведены в табл. 1.1. При рассмотрении данной таблицы следует учитывать, что тот или иной параметр в случае конкретного

анализатора может принимать одно или несколько из указанных в таблице значений.

Интерпретация заключается в отображении входного высказывания на знания системы. Основными задачами данного этапа являются: буквальная интерпретация высказывания в контексте диалога; интерпретация высказывания на намерения говорящего.

Буквальная интерпретация состоит в том, чтобы, учитывая контекст диалога, идентифицировать образы тех сущностей области интерпретации, которые имел в виду говорящий. В качестве области интерпретации могут использоваться: проблемная область; область языка общения (если высказывание пользователя касается языка общения); область системы (если пользователь интересуется возможностями и состояниями системы; интерпретация на эту область особенно важна при возникновении «непонимания» между пользователем и системой); область пользователя (если высказывание содержит сведения о знаниях или намерениях пользователя); область дискурса (если в высказывании содержатся ссылки на предыдущие или последующие высказывания).

Вторая задача интерпретации состоит в том, чтобы, применяя имеющиеся у системы методы вывода, определить, как обрабатываемое высказывание соотносится с целями и планами участников общения. Строго говоря, данная задача решается совместно диалоговым компонентом и компонентом понимания высказываний. Как отмечалось, одной из функций диалогового компонента является определение отношения входного высказывания к текущей цели. При выполнении этой функции текущая цель (т. е. ее описание в форме внутреннего представления) передается компоненту понимания, который пытается интерпретировать на нее входное высказывание. Решение данной задачи в общем случае представляет собой чрезвычайно сложную проблему. Пока даже не ясен механизм (модель) формирования текущих целей диалоговым компонентом. Положение усугубляется тем, что одно и то же высказывание может использоваться для достижения целей, относящихся к различным областям. Для исследования этой задачи разработано несколько экспериментальных систем, на которых остановимся в следующем параграфе.

Компонент генерации высказываний решает в соответствии с результатами, полученными остальными компонентами системы, две основные задачи: *генерация смысла*, т. е. определение типа и смысла выходного высказывания системы во внутреннем представлении; *синтез высказывания*, т. е. преобразование смысла в высказывание на естественном языке.

Первая из указанных задач является сложной и мало изученной. Тип высказывания зависит от состояния системы и результатов, полученных предыдущими компонентами. Так, если система должна генерировать ответ на вопрос, то необходимо определить по состоянию системы, будет ли ответ прямой или косвенный. Прямой ответ (т. е. по существу заданного вопроса) может быть дан только в том случае, если обработка вопроса завершилась успешно. Если в процессе обработки вопроса возникают какие-то затруднения, то более уместным может быть косвенный ответ, раскрывающий суть возникших затруднений и объясняющий невозможность прямого ответа.

В общем случае при решении задачи формирования смысла выходного высказывания необходимо учитывать прагматический аспект, т. е. цели участников общения. Однако в большинстве существующих систем данная задача решается с помощью достаточно простых подходов. В промышленных системах общения генерация смысла обычно заключается в редактировании значений атрибутов и (или) выборе шаблона ответа. В экспериментальных системах для выражения смысла генерируется полное семантическое представление, включающее одно или несколько связанных событий (понятий) с одним или несколькими исполнителями на каждую роль.

Вторая задача компонента генерации высказываний состоит в синтезе естественно-языкового выражения, соответствующего внутреннему представлению выходного высказывания. Данная задача подразделяется на этапы семантического, синтаксического и морфологического синтеза [Попов, 1982]. Сложность

задачи синтеза определяется требованиями к естественности и выразительной мощности выходных высказываний. Под *естественностью* обычно понимается степень близости к естественному языку, т. е. наличие таких свойств, как синонимия и омонимия слов и словосочетаний, свободный порядок слов и т. п. Под *выразительной мощностью* понимается возможность выразить разнообразные понятия, отношения, кванторы, процедуры и т. п. Естественность и выразительная мощност выходных высказываний в существующих системах могут быть различными. Например, высказывания могут фактически не синтезироваться, а выбираться из заранее заготовленного списка либо имеется шаблон ответа, в который подставляются некоторые слова, представляющие собой значения искомых атрибутов; при этом может использоваться морфологический синтез. Большая естественность достигается, если выходное высказывание формируется из семантического представления события (или понятия) с применением морфологии, синтаксиса (для определения порядка и формы слов) и семантики (для выбора лексем и обеспечения семантической сочетаемости слов в синтезируемом высказывании).

Для понимания принципов построения ЕЯ-систем важен также вопрос об используемых в системе знаниях, поскольку именно знания, представленные в различных формах, являются той базой, на которой осуществляется решение рассмотренных выше задач.

Знания ЕЯ-систем можно классифицировать по различным основаниям (см., например, [Попов, 1982; Попов и др., 1976; Представление, 1984]). Будем придерживаться следующей классификации (рис. 1.2). На верхнем уровне выделяются: собственно знания; способ представления знаний.

Собственно знания классифицируются по областям и по видам знаний. Наиболее существенными с точки зрения процесса ЕЯ-общения являются следующие области знаний: проблемная область; область языка; область системы; область пользователя; область диалога (дискурса). Разнообразие областей определяет множество возможных интерпретаций входных высказываний.

К основным видам знаний относятся факты (фактические знания) и операционные знания. Факты представляют собой возможные знания о сущностях, составляющих некоторую область знаний. Операционные знания составляют информацию о способах изменения фактических знаний. Иначе говоря, эти знания задают процедуры преобразования. Часто для обозначения этих знаний используется термин «процедурные знания», однако следует иметь в виду, что операционные знания могут быть представлены как в процедурной, так и в декларативной форме.

Способ представления знаний включает два аспекта: способ организации знаний и модель представления. Способы организации знаний различаются по уровням представления и уровням детальности. По уровням представления выделяют знания нулевого уровня (конкретные и абстрактные знания) и знания более высоких уровней (метазнания). Первый уровень составляют знания о том, как в системе представлены знания нулевого уровня. Число уровней представления может быть продолжено. Разделение знаний по уровням представления обеспечивает возможность гибкой настройки и адаптации ЕЯ-системы.

Организация знаний по уровням детальности позволяет рассматривать знания с различной степенью подробности. Количество уровней детальности зависит от специфики решаемых задач, количества знаний и способа их представления. Обычно выделяется не менее трех уровней, отражающих общую организацию знаний, логическую и физическую организацию частных структур знаний. Введение нескольких уровней детальности обеспечивает дополнительную гибкость системы, так как изолирует изменения одного уровня от других.

Модели представления знаний обычно подразделяются на логические и эвристические. Основные свойства и возможности этих классов моделей подробно рассмотрены в кн. 2 данного справочника. Здесь же лишь отметим, что модели представления знаний обычно классифицируют в зависимости от степени декларативности (процедурности) языковых средств, используемых для описания

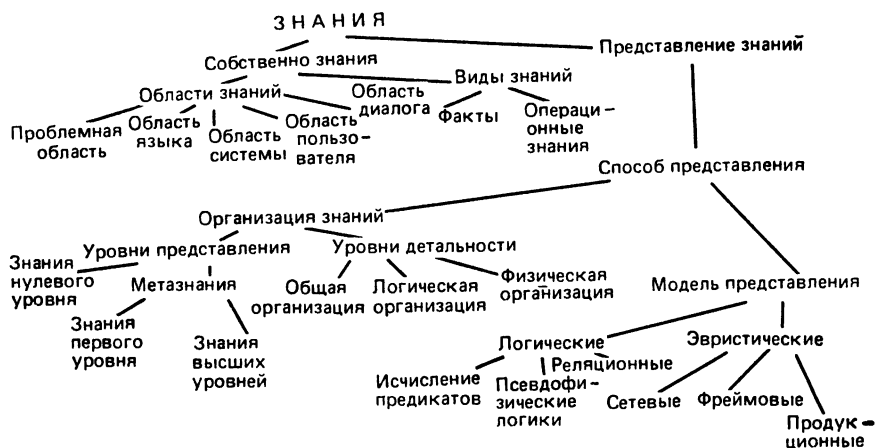


Рис. 1.2. Знания ЕЯ-систем

знаний. Различие между декларативным и процедурным представлением можно выразить как различие между «знать что» и «знать как». Каждое представление имеет свои достоинства и недостатки. Стремление наиболее полно использовать достоинства обоих видов представлений привело к разработке смешанных представлений, т. е. декларативных представлений с присоединенными процедурами (например, фреймворки модели и модели, использующие расширенные семантические сети).

1.3. Методы реализации ЕЯ-систем

Приведем методы реализации основных функциональных компонентов, получившие наиболее широкое распространение в практике создания ЕЯ-систем.

1.3.1. Диалоговый компонент. Диалог можно рассматривать на трех уровнях: общая (глобальная) структура, характеризующая тип диалога и класс решаемых задач; тематическая структура, отражающая структуру конкретной задачи; структура шага диалога (локальная структура), отражающая взаимодействие участников в элементарном акте диалога.

На уровне глобальной структуры действия ЕЯ-системы обычно задаются в виде последовательности этапов, определяемых в зависимости от класса решаемых задач. Так, в случае общения с экспертными системами глобальная структура включает следующие этапы: инструктаж; определение задачи; решение задачи; объяснения в ходе решения задачи; выдача результатов решения задачи и их оценка; объяснения после решения задачи; определение причин неудачи и приобретение новых знаний.

Перечисленные этапы не обязательно должны выполняться в каждом конкретном диалоге, но если они выполняются, то в том порядке, который указан в глобальной структуре. Тот или иной этап может не выполняться либо в связи с явным указанием пользователя, либо по умолчанию. Например, если результат решения задачи удовлетворяет пользователя, то этапы объяснения и приобретения могут быть пропущены. В силу простоты и статичности глобальной структуры она, как правило, встраивается в управляющий механизм диалогового компонента (т. е. задается процедурно); однако в ряде систем, ориентированных на многоцелевое применение (см., например, HAM-ANS [Хоерперг et al., 1983], глобальная структура задается декларативно с помощью правил, имеющих вид продукций.

Тематическая структура диалога обычно представляется в виде сценария, в рамках которого определяются: структура задачи, решаемой в процессе общения, т. е. разбиение задачи на упорядоченное множество подзадач; распределение подзадач между участками общения, т. е. определение, какие подзадачи решаются системой и какие — пользователем; языковые средства, используемые при обращениях к пользователю. Для задания тематической структуры (сценария) диалога в существующих системах общения применяются различные способы. Они могут быть сгруппированы по крайней мере в три класса: сценарий присутствует в системе в «готовом» виде (например, он встраивается в систему при ее создании или вводится в процессе настройки системы на проблемную область); сценарий генерируется системой в процессе решения задачи; некоторые компоненты сценария присутствуют в системе в готовом виде, а некоторые генерируются. Последний класс является композицией предыдущих, поэтому ограничимся рассмотрением способов, составляющих первый и второй классы.

Готовый сценарий может быть задан в виде частично упорядоченного множества правил с параметрами, значения которых устанавливаются в процессе решения конкретной задачи. В зависимости от значений параметров между правилами устанавливается отношение строгого порядка, определяющее тематическую структуру конкретного диалога. Использование готовых сценариев целесообразно в тех случаях, когда к системе предъявляются жесткие требования по быстродействию (это характерно для промышленных и коммерческих систем), а решаемые задачи имеют устойчивую структуру и заранее известны функции (роли) участников общения. Класс подобных задач весьма широк. Это большинство задач общения с базами данных [Дракин и др., 1988], задачи предупреждения или устранения определенных видов локальных неудач [Городецкий и др., 1985], задачи, возникающие при настройке ЕЯ-систем на проблемную область [Grosz et al.; 1987, Moser, 1984]. Во всех случаях функция ведения диалога ограничивается интерпретацией готового сценария. При этом если сценарий не встроен заранее в диалоговый компонент, а должен вводиться в систему при ее настройке на проблемную область, то для его описания используются специальные языки. Примерами отечественных систем, имеющих языки описания сценариев, могут служить АИСТ [Дракин и др., 1988] и АДС [Брябрин, 1981].

Если сценарий диалога не присутствует в системе в готовом виде, а генерируется в процессе решения задачи, то в диалоговый компонент включается специальный механизм вывода (планирования). Метод планирования определяется в зависимости от используемой в конкретной ЕЯ-системе системы представления знаний. Например, если для представления знаний применяется исчисление предикатов, то процесс генерации сценария реализуется методами доказательства теорем. Методы планирования для различных систем представления знаний подробно рассматриваются в кн. 2 данного Справочника, поэтому здесь их классификация не приводится. Отметим лишь, что специфика применения методов планирования для генерации сценариев состоит в том, что в качестве операторов, решающих элементарные подзадачи, рассматриваются не только действия, ведущие к изменению отношений между сущностями проблемной области, но и типовые действия, которые соответствуют определенным речевым поступкам (речевым актам [Падучева, 1982а]), например таким, как «сообщать», «спрашивать», «предлагать». Это позволяет естественным образом включить в план решения задачи — в последовательность действий по решению элементарных подзадач, полученную с помощью механизма вывода, речевые акты, т. е. определить языковые средства, используемые для обращений к пользователю.

Генерация сценариев диалога целесообразна в тех случаях, когда структура задачи зависит от контекста ситуации, в которой происходит ее решение, а полный перечень ситуаций не может быть определен заранее. К подобным задачам относится большинство задач общения с экспертными системами [Попов, 1987], а также многие задачи, возникающие при обработке связанных тек-

Т а б л и ц а 1.2.

Основные типы шагов диалога

Инициатор шага	Вид действия	Вид реакции	Способ спецификации подзадач	
			ограниченность пространства выбора	однозначность определения
1. Пользователь	Задание	Отчет или диагностическое сообщение	Неограниченное	Неоднозначное
2. Пользователь	Команда	Выполнение команды или диагностическое сообщение	Ограниченное	Однозначное
3. Система	Простой вопрос	Ответ на простой вопрос	Фиксированное	То же
4. Система	Вопрос с фиксированной структурой ответа	Ответ в заданной структуре	Ограниченное	»
5. Система	Вопрос со свободной структурой ответа	Произвольный ответ	Неограниченное	Неоднозначное
6. Система	Альтернативный вопрос (предложение выбора)	Выбор альтернативы	Ограниченное	Однозначное

стов [Winograd, 1983]. (В последнем случае методы планирования позволяют устанавливать цели и строить планы действий (в том числе и речевых) участников событий, описываемых в текстах.)

При задании локальной структуры (см. § 1.1) шаг диалога состоит из действия и реакции и характеризуется следующими параметрами: инициатор и тип инициирования (вид действия); способ влияния действия на реакцию; способ спецификации подзадачи, решаемой на данном шаге. Последний параметр, следуя [Деннинг и др., 1984], будем характеризовать двумя подпараметрами: ограниченностью пространства выбора функции (и/или параметров), используемой для решения подзадачи, и однозначностью определения функции. Данные параметры позволяют выделить шесть основных типов шагов диалога для ЕЯ-систем (табл. 1.2).

На локальном уровне диалога задача диалогового компонента состоит в определении параметров текущего шага. Инициатором шага в общем случае может быть как пользователь, так и система. Действия пользователя подразделяются на задания и команды. *Задания* предусматривают формулировку решаемой задачи (подзадачи) и ввод необходимых для ее решения параметров. При этом однозначность определения функции, обеспечивающей решение этой задачи, не гарантируется. Например, пользователь может специфицировать задачу, для решения которой в системе отсутствует подходящая функция (либо такая функция есть, но она не может быть выполнена на данном шаге), или задать значения параметров не в том виде в котором требуется системе. *Команды* обычно служат для перехода инициативы или для выполнения технологических действий, например листания отчета, получения твердой копии и т. п. Фиксированность набора команд позволяет системе легко идентифицировать команду и однозначно определять требуемую для выполнения команды функцию.

Если инициатива принадлежит системе, то вид действия определяется исходя из того, насколько диалоговому компоненту известна функция (и/или параметры), с помощью которой осуществляется решение подзадачи на данном шаге. Если функция известна, а неизвестны лишь некоторые параметры этой функции, то используется *простой вопрос*, например: «Введите номер отдела, в котором работает Петров».

Альтернативные вопросы и вопросы с фиксированной структурой ответа применяются в тех случаях, когда на данном шаге возможно выполнение одной из нескольких функций, а выбор осуществляется в зависимости от реакции пользователя. Альтернативные вопросы ограничивают выбор явно, т. е. пользователю предъявляется множество возможных ответов. Вопросы с фиксированной структурой ответа предусматривают неявное определение функции. Эти вопросы содержат анкету для ввода ответа. *Анкета* состоит из именованных полей, предназначенных для ввода соответствующих значений. Определение функции осуществляется в зависимости от того, какие поля анкеты будут заполнены пользователем при вводе ответа.

Вопросы со свободной структурой ответа не содержат никаких сведений относительно ожидаемых свойств ответа. В принципе эти вопросы в существенно меньшей степени ограничивают (вплоть до отсутствия каких-бы то ни было ограничений) область выбора функции и параметров. Естественно, что при этом не гарантируется однозначность их определения. С этой точки зрения вопросы со свободной структурой ответа аналогичны заданиям, разница между ними — только в инициаторе действия.

Из рассмотрения основных типов шагов диалога следует, что при действиях, не ограничивающих возможные реакции (заданиях и вопросах со свободной структурой ответа), не гарантируется однозначная спецификация задачи. Кроме того, неоднозначность спецификации может иметь место и в тех случаях, когда действие системы ограничивает реакцию пользователя (простой вопрос и вопрос с фиксированной структурой ответа), но вследствие использования естественного языка ответ неправильно понимается системой. Во всех по-

добных случаях требуется приведение ситуации к однозначной. На практике для этого применяется перефразирование или изменение темы диалога.

Перефразирование заключается в переформулировании на естественном языке высказывания (задания или ответа) пользователя. При неоднозначном понимании все возможные (с точки зрения системы) варианты выдаются в виде альтернативного вопроса. Частным случаем перефразирования является «эхо». При этом повторяется часть высказывания пользователя, в которой система сомневается.

При изменении темы диалога текущий шаг диалога прерывается, и для достижения однозначного понимания внутри него создается поддиалог из одного или нескольких шагов. Поддиалог может быть заранее запланирован в сценарии диалога либо инициирован в результате перехвата инициативы ранее пассивным участником. В последнем случае в зависимости от того, кто из участников осуществляет перехват инициативы, диалоговый компонент либо формирует перехват, либо обрабатывает его.

Как правило, действия участников по перехвату инициативы ограничены моментом перехвата, способом перехвата и целями, которые участники могут преследовать, перехватив инициативу. Обычно перехват инициативы разрешается в те моменты, когда активный участник находится в ожидании реакции пассивного участника. Наиболее распространенный способ перехвата инициативы системой — простые и альтернативные вопросы. В первом случае целью системы является уточнение параметра, а во втором — функции, которая должна быть выполнена для решения подзадачи на прерванном шаге диалога. Для перехвата инициативы пользователем обычно применяются специальные стандартные команды, смысл которых заранее известен системе. Появление в высказывании пользователя подобной команды сигнализирует системе как о наличии перехвата инициативы, так и о цели перехвата.

Если способ перехвата инициативы пользователем не ограничен специальными командами, то диалоговый компонент должен определять по смыслу очередного высказывания его отношение к текущей цели (теме). Если взаимосвязь высказывания с текущей целью установить не удастся, то это высказывание должно рассматриваться как перехват инициативы. При этом возникает достаточно сложная задача определения цели, которую, перехватив инициативу, намерен преследовать пользователь. Данная задача в настоящее время еще не имеет удовлетворительного решения, однако следует подчеркнуть, что способность диалогового компонента обрабатывать перехваты инициативы с учетом целей участников является одним из необходимых условий для организации гибкого диалога, гарантирующего достижение в процессе общения глобального успеха.

1.3.2. Компонент понимания высказываний. Понимание высказываний, как отмечалось в § 1.2, включает анализ и интерпретацию.

В методах анализа обычно выделяются анализ слов, предложений и текстов. *Анализ слов* сводится к морфологическому анализу и обнаружению и исправлению орфографических ошибок. Цель *морфологического анализа* состоит в получении основ (под основой понимается словоформа с отсеченным окончанием) со значениями грамматических категорий (например, часть речи, род, число, падеж) для каждой из словоформ высказывания, поступившего на вход ЕЯ-системы. Методы морфологического анализа были детально разработаны еще при создании первых ЕЯ-систем. Примерами отечественных ЕЯ-систем с достаточно полной для практических потребностей реализацией морфологического анализа могут служить ПОЭТ [Попов, 1982], TULIPS [Мальковский, 1985] и АИСТ [Дракин и др., 1988].

Методы обнаружения и исправления орфографических ошибок подразделяются на два класса в зависимости от того, используют они словари основ или нет. К методам, не использующим словари, относятся частотные и полиграммные. *Частотные методы* основаны на сортировке слов по частоте их встречаемости в текстах. Предполагается, что частота встречаемости слов, содержащих ошибки, низкая. Однако низкая частота встречаемости и у правильных, но ред-

ко встречающихся слов, что значительно снижает эффективность частотных методов [Pollack et al., 1984]. В *полиграммных методах* [Riseman et al., 1977] для поиска ошибок применяют списки возможных сочетаний букв в словах (обычно анализируются пары и тройки идущих подряд букв). Полиграммными методами целесообразно пользоваться в системах с открытым (пополняемым) словарем наряду с методами, основанными на словарях.

Методы, в которых используются словари, разделяются в зависимости от типа применяемой стратегии на абсолютные и относительные. К *абсолютным* относится «исторический» метод, основанный на словаре встречаемых ранее ошибок. Данный метод реализован, например, в системе SPEEDCOP [Pollack et al., 1984]. Эффективность исторического метода существенно зависит от размера текстов, на основе которых порожден словарь ошибок. Относительный метод состоит в нахождении в словаре таких слов, которые наиболее похожи на искаженное слово, и выборе среди них правильного. Обычно искаженное слово подвергается определенной обработке для получения из него правильных слов. Обработка, как правило, включает действия по пропуску, перестановке и вставке букв (см., например, [Белоногов и др., 1984; Pollack et al., 1984]). При этом для уменьшения списка новых слов применяются частотные и полиграммные методы.

Анализ предложений обычно сводится к синтаксическому и семантическому анализу, выполняемому отдельным функциональным блоком-анализатором (parser). Наиболее распространенные методы анализа предложений, так же как и методы морфологического анализа, были разработаны еще при создании первых ЕЯ-систем и предназначались для обработки только «правильных», т. е. не содержащих отклонений от грамматической нормы предложений. Обычно при описании анализаторов (см., например, [Ангелова и др., 1984; Виноград, 1976; Попов и др., 1976]) основное внимание уделяется распределению функций между синтаксическим и семантическим анализом и порядку их выполнения. Однако с точки зрения современных требований к ЕЯ-системам (см. § 1.1) более важным является вопрос о том, насколько существующие анализаторы могут быть приспособлены к обработке «неграмматичностей», т. е. характерных для диалогов между людьми высказываний с отклонениями от грамматической нормы (лексические и грамматические ошибки, пропуски, повторы, шумы, эллипсис, идиомы и т. п.). Сравним по этому параметру следующие типы анализаторов: традиционные; концептуальные; анализаторы, использующие сопоставление по образцам; анализаторы, использующие разнообразные стратегии.

Традиционные анализаторы. Наиболее распространенным способом анализа ЕЯ-предложений является разбор сверху вниз, слева направо, основанный на некоторой фиксированной грамматике. В последние годы подобные методы обычно выполнялись с применением АТН-техники, т. е. с помощью расширенных сетей переходов [Вудс, 1978]. Такие анализаторы осуществляют разбор предложения либо в общих грамматических категориях (см., например, LUNAR [Woods et al., 1978]), либо в терминах категорий, имеющих значение в некоторой ограниченной области (см., например, SOPHIE [Brown et al., 1979], МИВОС [Преображенский и др., 1979]). Анализаторы этого типа чрезвычайно «хрупки», т. е. они терпят неудачу при разборе предложений, содержащих минимальные отклонения от нормы.

Хрупкость традиционных анализаторов обусловлена тем, что их алгоритм осуществляет поиск сверху вниз среди разборов, допускаемых грамматикой, того разбора, который соответствует обрабатываемому предложению. Если некоторый частный разбор при сопоставлении ему очередного слова противоречит используемой грамматике, то для анализатора это сигнал того, что на более раннем этапе поиска сделан ошибочный выбор. Таким образом, неудача на некотором шаге разбора является сигналом для выбора очередного из возможных разборов, т. е. принципиальные затруднения возникают при обработке предложений, содержащих отклонения от грамматики.

Для преодоления указанных недостатков традиционных анализаторов были предложены способы, позволяющие ослаблять действие грамматических

правил (см., например, [Weischedel et al., 1980]). Однако это возможно только в ограниченном классе грамматических отклонений. Кроме того, предпринимались попытки добавить в ATN-сеть дополнительные специфические дуги, которые имеют дело с проблематичными входными предложениями. Некоторые из этих дуг выполняли функцию сопоставления по образцу (см. ниже). Тот факт, что для обработки неграмматичных предложений приходится осуществлять радикальные преобразования ATN-техники, говорит о ее малой пригодности для обработки высказываний, имеющих место в реальных, естественных диалогах [Hayes et al., 1983].

Один из возможных подходов к преодолению хрупкости традиционных анализаторов состоит в одновременном применении нескольких подграмматик. Каждая из подграмматик предназначена для анализа частных конструкций какого-либо одного вида. Применение подграмматик осуществляется независимо, поэтому неудача одной подграмматики не влияет на возможности других. Впервые подобный подход был реализован в системе PLANES [Waltz, 1978], которая имеет подграмматики для каждого типа известных систем сущностей. При данном подходе предложение в процессе разбора разбивается на несколько независимых фрагментов. В этом случае в задачу анализатора входит построение общей (объединенной) интерпретации предложения. Если проблемная область достаточно ограничена (как это имеет место в системе PLANES), то интерпретация фрагментов всегда уникальна, однако в общем случае эта задача не имеет единственного решения и может стать трудноразрешимой.

Концептуальные анализаторы. Анализаторы данного типа используют методы разбора, направляемые значениями базовых событий, обнаруженных в анализируемых предложениях. Наиболее известными разновидностями данного подхода являются анализаторы, основанные на модели концептуальной зависимости [Шенк, 1980] и на модели управления [Апресян, 1975]. Анализатор первого типа был впервые реализован в системе MAPGE [Шенк, 1980], а второго — в системе ПОЭТ. Концептуальные анализаторы не разрабатывались специально для анализа неграмматичных предложений. Однако заложенные в них идеи в принципе позволяют этим алгоритмам работать в условиях пропусков и повторов слов. Такие системы, как FRUMP [De Jong, 1979], IPP [Шенк и др., 1983], RESEARCHER [Lebowitz, 1985], SNUKA [Pazzani, 1983] с концептуальными анализаторами по сути обладают иммунитетом к ошибкам, так как они игнорируют непонятные им слова, а понятные приспособляют (даже если в них есть ошибки) к базовым событиям обрабатываемого предложения.

Анализаторы, использующие сопоставление по образцам. Анализаторы данного класса основаны на том, что в простейшем случае анализ сводится к сопоставлению предложения с некоторым множеством образцов, представляющих собой последовательности из одного или нескольких слов. Подобные анализаторы широко применялись в ранних ЕЯ-системах. Многие методы анализа, основанные на сопоставлении по образцам, содержат в образце не только константы, но и переменные. При этом предполагается, что переменные образца могут сопоставляться с любой строкой символов. Гибкость анализаторов определяется гибкостью процесса сопоставления. Различаются следующие формы сопоставления: синтаксическое, параметрическое, семантическое и принуждаемое (см., например, [Попов, 1982]). Разнообразие форм сопоставления позволяет анализировать входные предложения, отклоняющиеся от традиционной грамматики в произвольной степени, однако глубина проникновения в смысл (см. § 1.2) подобных анализаторов обычно невелика.

Возможности методов анализа, основанных на сопоставлении по образцам, можно увеличить, используя частичное сопоставление предложения с одним или несколькими образцами. Примером такой системы может служить Flex-P [Hayes et al., 1981b]. Следует отметить, что возможности частичного сопоставления исследованы мало. Однако бесспорно, что этот метод анализа весьма эффективен при обработке предложений с отклонениями от грамматической нормы. Кроме того, сопоставление по образцам позволяет успешно обрабаты-

вать идиомы. Так как идиомы не могут быть обработаны путем интерпретации образующих их слов, то для их обработки неизбежно привлечение механизмов, подобных сопоставлению по образцам. По этой причине некоторые системы, например LUNAR, основанные на традиционных методах анализа, для обработки идиом включают стадию преданализа, на которой применяется механизм сопоставления по образцам.

Методы анализа, использующие сопоставление по образцам, имеют определенные ограничения. Если все предложение сопоставляется с одним образцом, то ясно, что такой подход не позволяет обеспечить точную обработку сложных предложений. Если предложение сопоставляется с несколькими образцами, каждый из которых подходит к отдельному фрагменту предложения, то возникает задача определения границ фрагментов. Кроме того, после сопоставления фрагментов одного предложения с несколькими образцами возникает задача получения общей интерпретации всех фрагментов данного предложения.

Анализаторы, использующие разнообразные методы. Все анализаторы, относящиеся к рассмотренным выше классам, основываются на каком-либо одном методе. Недавние исследования [Hayes et al., 1981a] показали, что использование в одном анализаторе нескольких специфических методов позволяет обеспечить гибкость процесса анализа, необходимую для обработки неграмматичных конструкций. Примерами таких анализаторов могут служить CASPAR [Carbonell et al., 1981; Hayes et al., 1981a] и MULTIPAR [Minton et al., 1985]. В настоящее время данный подход еще не получил такого распространения, как рассмотренные ранее подходы. Это объясняется, на наш взгляд, его сравнительной новизной и недостаточной исследованностью.

Перейдем к описанию методов анализа связного текста (дискурса). Связность дискурса достигается как лингвистическими средствами, имеющими языковое выражение, так и экстралингвистическими (ситуационными) средствами — «умолчаниями», не имеющими языкового выражения и основанными на общности знаний участников общения о цели общения и проблемной области. На этапе анализа связного текста, как правило, решается задача выявления связей между предложениями, выражаемых лингвистическими средствами, а на этапе интерпретации — ситуационными.

К основным лингвистическим средствам связи предложений относятся ссылки и эллипсис. В проблеме установления ссылок могут быть выделены две задачи: поиск в предыдущих предложениях (контексте) сущности (референта), обозначающей данной ссылкой; определение соответствия между референтом и ссылкой. Простейшим методом решения первой задачи является поиск референта в заданном количестве предыдущих предложений [Виноград, 1976; Norman et al., 1975]. Однако отсутствие критерия для определения количества просматриваемых предыдущих предложений приводит на практике как к увеличению времени поиска, так и к ошибкам в установлении ссылок. Решение второй задачи является тривиальным для простейших видов ссылок (т. е. в случае тождества референта и ссылки) и весьма трудным для случаев несоответствия референта и ссылки. Отсутствие удовлетворительных методов решения обеих задач в общей постановке на этапе анализа текста стимулировало попытки их решения на этапе интерпретации (см. ниже).

Задача обработки эллиптических конструкций решается на этапе анализа также в ограниченной постановке. Под *эллипсисом* понимается сжатая форма высказывания, смысл которой определяется либо предыдущими высказываниями (текстовый эллипсис), либо ситуацией, имеющей место в проблемной области (ситуативный эллипсис) [Падучева, 1982б]. С формально-грамматической точки зрения высказывания, содержащие эллипсис, выглядят как неполные (т. е. содержащие пропуски слов) предложения. На этапе анализа может быть обработан (т. е. восстановлен) только текстовый эллипсис. Сущность методов восстановления текстового эллипсиса (см., например, [Попов, 1982; Hendrix, 1977; Hoerppner et al., 1983]) состоит в подстановке фрагментов предыдущих высказываний в текущее высказывание, содержащее эллипсис. При этом про-

пущенные слова в текущем высказывании как бы восстанавливаются из текста предыдущего высказывания. Восстановление ситуационного эллипсиса осуществляется на этапе интерпретации (см. § 1.2).

На этапе интерпретации решаются две основные задачи: буквальная интерпретация высказываний в контексте диалога; интерпретация на цели участников общения. Методов решения этих задач в общей постановке не существует. Однако применительно к простым проблемным областям их решение существенно упрощается.

К простым проблемным областям относятся различные области, над которыми решаются простые задачи информационного обслуживания, например: справочные задачи (о погоде, о товарах, о литературе), задачи резервирования (мест, билетов, товаров) и т. п. Все эти задачи обладают «прозрачной» структурой, т. е. они оперируют ограниченным множеством сущностей, которые являются параметрами предлагаемого вида обслуживания. Структура простых задач информационного обслуживания соответствует структуре фрейма. При этом параметры обслуживания соответствуют слотам фрейма, а весь вид обслуживания представляется некоторым фреймом. Работа ЕЯ-системы при обслуживании пользователя состоит в заполнении (означивании) слотов и, возможно, в выполнении некоторых манипуляций над фреймами с заполненными слотами.

Для решения обеих задач интерпретации в рамках фрейм-представлений используется единый механизм означивания фреймов. При этом структура целей участников общения (т. е. разбиение целей на подцели) определяется структурой фрейма, описывающего данную задачу, а подцели состоят в заполнении слотов, т. е. в идентификации сущностей через описания, представленные в результатах анализа входных высказываний.

В общем случае процесс идентификации некоторой сущности может иметь три исхода [Hayes et al., 1983]: однозначный — данному описанию сопоставляется единственная сущность; многозначный — описанию сопоставляется более чем одна сущность; неудовлетворительный — описанию не сопоставляется ни одна сущность. Последние два исхода рассматриваются как неудачи буквальной интерпретации и служат сигналами о необходимости установления подцелей более глубокого уровня, предусматривающих устранение неудач. При этом в диалоговый компонент кроме сообщения о неудаче и типе неудачи передаются исходные данные, позволяющие сформировать (с помощью компонента генерации высказываний) действие системы по перехвату инициативы и открытию уточняющего поддиалога, преследующего новую подцель.

В случае многозначной интерпретации вместе с сообщением о неудаче передаются возможные варианты сопоставления, что позволяет системе генерировать альтернативный вопрос. При неудовлетворительной интерпретации от компонента понимания требуется указать причины, по которым рассматриваемое описание является неудовлетворительным. В большинстве случаев (см., например, [Городецкий и др., 1985; Kaplan et al., 1979]) причины неудовлетворительной интерпретации сводятся к неудовлетворительным (неполным или неоднозначным) результатам анализа входных высказываний и к несоответствию presupпозиций пользователя знаниям системы.

При решении задач интерпретации важную роль играет имеющееся в системе представление общей точки зрения на то, о чем идет речь в текущий момент. Эту точку зрения часто называют *фокусом* (или фокусом внимания) [Grosz, 1977]. Разделяемый участниками фокус позволяет им повысить компактность диалога за счет того, что сущности, находящиеся в фокусе, могут либо вообще не упоминаться в высказывании (эллипсис), либо упоминаться в виде кратких описаний (ссылок). Покажем, как используется фокус для установления референтов ссылок и восстановления эллипсиса.

В работе [Hayes et al., 1983] предлагается приравнять фокус к текущей подцели, преследуемой участниками общения. Подобная трактовка рассматривает фокус как набор сущностей, на которые в текущий момент направлено внимание участников, при этом текущая подцель может быть определена как

множество сущностей, соответствующих этой подцели. Ясно, что данная трактовка фокуса обеспечивает решение задач установления референтов ссылок и восстановления эллипсиса. Поясним это на примере следующего диалога:

П1: Какая зарплата у Петрова?

С1: Вы имеете в виду Алексея Петрова или Николая Петрова?

П2: Я имею в виду Николая.

В данном диалоге пользователь ссылается на Николая Петрова (П2). Референт этой ссылки может быть легко идентифицирован. Действительно, высказывание П1 генерирует подцель «заполнить слот личность». Вследствие многозначной интерпретации система генерирует подцель «выбрать между двумя личностями». Ответ пользователя выбирает одну из них. С таким же успехом пользователь мог бы ответить: «Я имел в виду первого из них». Заметим, что приведенные рассуждения опираются на тот факт, что высказывание пользователя (П2) не изменило текущую подцель и, следовательно, не изменило текущий фокус.

Если бы в приведенном выше диалоге ответ пользователя (П2) содержал эллипсис, например ответ имел бы вид «Николая» или «Первого из них», то эллипсис мог бы быть легко восстановлен. Действительно, если предположить, что пользователь согласен с фокусированной подцелью системы (выбрать между двумя личностями), то естественно считать, что сущность, упомянутая в ответе, определяет выбор пользователя. Отметим, что при данном методе не возникает необходимость достраивать эллиптическое высказывание до полного высказывания (как это требуется при восстановлении эллипсиса на этапе анализа).

Еще раз подчеркнем, что изложенные выше методы базируются на фрейм-представлениях и широко применяются в ЕЯ-системах, ориентированных на решение простых задач информационного обслуживания. Что же касается методов интерпретации, используемых в более сложных проблемных областях (например, понимание связанных текстов, описывающих разворачивающиеся во времени события со многими действующими лицами), то они находятся в стадии становления и пока не поддаются обобщенному описанию, так как сильно зависят как от условий конкретных задач, решаемых ЕЯ-системой, так и от специфики применяемых средств представления знаний.

1.3.3. Компонент генерации высказываний. Процесс генерации высказываний состоит из генерации смысла высказывания и синтеза высказывания на ЕЯ. Первый этап часто называется (см., например, [Макьюин, 1986; Попов, 1982]) вневингвистическим (или концептуальным) синтезом, а второй — лингвистическим синтезом. Результатом выполнения первого этапа является внутреннее представление смысла генерируемого высказывания. Как правило, на первом этапе решаются следующие задачи: определение информации, которая должна быть сообщена пользователю; определение уровня общности информации, включаемой в высказывание; выделение из множества аспектов, описывающих сущности, о которых говорится в высказывании, аспектов, интересующих и понятных пользователю; разбиение сообщаемой информации на части, соответствующие будущим предложениям, и установление последовательности этих частей; определение лексем и построение семантического представления высказывания. На втором этапе обычно решаются следующие основные задачи: построение синтаксической структуры отдельных предложений; приписывание морфологической информации вершинам синтаксических структур отдельных предложений; морфологический синтез словформ.

До последнего времени в большинстве исследований по генерации высказываний на ЕЯ наибольшее внимание уделялось проблемам, связанным с решением задач второго этапа. В ходе этих исследований были развиты методы прямой трансляции формализованного представления смысла в предложения на английском языке [Chester, 1976; Simmons et al., 1972], разработаны системные грамматики и механизмы их употребления для порождения текста [Kau, 1979; Mann, 1983], предложены критерии для отбора слов из общего словаря системы. Для генерации высказываний на русском языке были разра-

ботаны и внедрены в ряд ЕЯ-систем и систем машинного перевода методы морфологического и синтаксического синтеза [Апресян и др., 1978; Козьмина и др., 1972; Кулагина, 1979; Попов и др., 1978].

По сравнению с задачами второго этапа, для решения которых существуют апробированные методы, задачи, решаемые на первом этапе генерации, изучены гораздо меньше. Поэтому в большинстве действующих ЕЯ-систем решение задач генерации смысла осуществляется, как правило, в упрощенной или сильно ограниченной постановке, что приводит к упрощению, а иногда и к вырождению задач синтеза высказываний на ЕЯ.

Наиболее простым методом генерации высказываний является метод, который основан на использовании заранее заготовленных шаблонов, содержащих текст на ЕЯ и переменные, вместо которых подставляются конкретные данные (описания сущностей). Как показала практика, данный метод особенно удобен для генерации высказываний типа диагностического сообщения, простого вопроса, альтернативного вопроса и вопроса с фиксированной структурой ответа в определенных стандартных ситуациях, не зависящих от проблемной области, в которой используется ЕЯ-система. В подобных ситуациях текущей целью системы является предупреждение или устранение типовых локальных неудач, вызванных, например, такими факторами: наличие в высказывании пользователя неизвестных системе слов; несоответствие presuppositions в высказывании пользователя знаниям системы; многозначная или неудовлетворительная интерпретация высказывания пользователя и т. п.

При методе шаблонов генерация высказываний осуществляется, строго говоря, совместно с диалоговым компонентом, который идентифицирует стандартную ситуацию (т. е. генерирует соответствующую этой ситуации цель) и определяет, какая информация (т. е. описания каких сущностей) должна быть включена в высказывание. Компонент генерации высказываний в этом случае выбирает подходящий формат для представления высказывания и конкретизирует его соответствующими данными, возможно, производя при этом морфологический синтез словформ.

Аналогичный метод может применяться и при генерации большинства отчетов, т. е. высказываний, выражающих результаты решения задач, полученные ЕЯ-системой. Отличие состоит в том, что вместо заранее заготовленных шаблонов используются шаблоны, которые порождаются компонентом генерации высказываний на основе метазнаний, т. е. описаний данных, включаемых в ответ. Заметим, что в ряде случаев, например в системах ЕЯ-общения с БД, результатом решения задачи часто является множество данных, формат представления которых определяется средствами генерации отчетов СУБД. Очевидно, что в этих случаях необходимость в компоненте генерации высказываний ЕЯ-системы вообще не возникает.

Генерация высказываний, которые не могут быть описаны с помощью шаблонов, представляет собой более сложную и недостаточно исследованную задачу. К таким высказываниям, в частности, относятся объяснения, генерируемые в качестве ответов на вопросы, областью интерпретации которых являются абстрактные знания и метазнания системы (см. § 1.2). Такие вопросы часто задаются неопытными конечными пользователями для того, чтобы получить общее представление о знаниях и возможностях ЕЯ-системы, например: «Какого рода данные содержатся в базе данных?», «Что такое издержки производства?», «Какая разница между общезаводскими накладными расходами и издержками производства?» и т. п. Основная особенность подобных вопросов заключается в том, что они не дают точного представления о необходимой информации в ответе. Как правило, содержание ответа (текста объяснения) зависит от ситуации, в которой задан вопрос, а зачастую и от познаний пользователя в данной области.

Для генерации объяснений, т. е. ответов на вопросы метауровня и абстрактного уровня, требуется достаточно тонкая классификация целей создания ЕЯ-текстов. Организация текста (т. е. информация, сообщаемая пользователю, уровень ее общности, разбиение на части, соответствующие будущим предло-

жениям и т. п.) определяется так называемыми дискурсными целями, такими, как, например: дать определение, описать, сравнить, подтвердить, уточнить и т. д. Каждой дискурсной цели может быть сопоставлен определенный способ организации текста, называемый дискурсной стратегией. Предполагается, что, выявив дискурсные стратегии, используемые людьми, и выразив их в подходящем формальном представлении, можно получить достаточно надежный и эффективный механизм генерации смыслов порождаемых текстов.

Один из возможных методов представления дискурсных стратегий применяется в системе TEXT [Макьюин, 1986]. Данная система генерирует объяснения в виде отдельных абзацев и может отвечать на вопросы следующих типов: вопросы на определения (дефиниции) объектов; вопросы о различии между объектами; вопросы на описания объектов; вопросы относительно имеющейся в базе данных информации. Дискурсные стратегии построения объяснений были получены эмпирически в результате изучения коротких определений и сопоставлений в различных справочниках и энциклопедиях. Каждая дискурсная стратегия представляется в системе TEXT в виде грамматики (дерева составляющих), терминалами которой служат так называемые риторические предикаты (например, идентификация объекта как члена какого-то класса, представление свойств (атрибутов) объекта и т. п.), определяющие тип информации из системной модели данных, которая может быть использована для их конкретизации.

Изложенные выше методы не претендуют на универсальность. Отсутствие общих методов решения задач генерации (в основном это касается задач генерации смысла высказываний) объясняется как сложностью и недостаточной изученностью проблемы, так и тем обстоятельством, что в силу исключительной способности человека понимать «плохо построенные» ответы наибольшее внимание при разработке ЕЯ-систем уделяется, как правило, компоненту понимания высказывания и диалоговому компоненту.

1.4. Настройка ЕЯ-систем

1.4.1. Гибкость ЕЯ-систем. Рассмотренные в предыдущем параграфе методы позволяют создавать ЕЯ-системы, обеспечивающие эффективное удовлетворение информационных потребностей конечных пользователей относительно ограниченных проблемных областей. Ограниченность проблемной области имеет принципиальное значение, так как позволяет наложить вполне «естественные» для каждой из проблемных областей ограничения на лексику, синтаксис, семантику и прагматику языка общения. Это дает возможность рассматривать процесс ЕЯ-общения не во всем многообразии и сложности, а в более узкой (и, следовательно, более простой) постановке. Вместе с тем, чтобы обеспечить общение относительно различных проблемных областей, каждая из которых характеризуется собственными, отличными от других областей ограничениями, а также проблемных областей, изменяющихся во времени, ЕЯ-система должна допускать возможность изменения (модификации) своих знаний, зависящих от специфики проблемной области, в которой функционирует система.

Процесс извлечения знаний из некоторого источника знаний и передачи их ЕЯ-системе называют *приобретением знаний*. В качестве источника знаний может выступать как человек, так и текст, в котором содержатся сведения о проблемной области. Если знания ЕЯ-системы не могут приобретаться без участия ее разработчиков (или программистов), то такие системы называются *жесткими*. Очевидно, что жесткие ЕЯ-системы не удовлетворяют требованиям, предъявляемым к средствам общения конечных пользователей (см. § 1.1).

Гибкие ЕЯ-системы допускают возможность приобретения знаний без привлечения программистов или разработчиков. Степень гибкости ЕЯ-системы определяется, по крайней мере, следующими факторами: составом (номенклатурой) приобретаемых знаний; пределами, в которых допускается модификация

знаний; инициатором процесса приобретения знаний (пользователь или система); требованиями к уровню профессиональной подготовки лиц, участвующих в процессе приобретения знаний.

Возможности приобретения знаний в гибких ЕЯ-системах основаны на использовании различных способов представления проблемно-независимых и проблемно-ориентированных знаний. Предполагается, что знания ЕЯ-системы (см. § 1.2) могут быть разделены на проблемно-независимые, используемые в любом из возможных приложений, и проблемно-ориентированные, определяемые спецификой конкретных приложений. Проблемно-независимые знания вводятся в ЕЯ-систему при ее разработке. Большая их часть представляется процедурно (т. е. в виде программ). Поэтому они не могут модифицироваться без участия программистов или разработчиков системы. Проблемно-ориентированные знания представляются декларативно (т. е. в виде фреймов, семантических сетей, выражений на языке исчисления предикатов, продукций и т. д.). Особенности различных способов декларативного представления знаний подробно рассматриваются в кн. 2 данного Справочника. Наиболее важное общее свойство декларативных представлений состоит в том, что механизм интерпретации декларативных знаний (т. е. программы ЕЯ-системы) не зависит от содержания этих знаний. Поэтому модификация декларативно представленных проблемно-ориентированных знаний может осуществляться без участия программистов или разработчиков системы.

Таким образом, состав приобретаемых знаний, а также пределы, в которых допускается модификация знаний, в каждой ЕЯ-системе зависят от масштабов использования в ней декларативно представленных знаний. Следует отметить, что применение декларативных представлений ведет в общем случае к ухудшению временных и объемных характеристик ЕЯ-системы. Поэтому при разработке конкретных ЕЯ-систем, особенно ориентированных на промышленное или коммерческое использование, часто приходится идти на компромиссы, т. е. искусственно «уменьшать» гибкость (за счет сокращения состава декларативно представленных знаний) для получения приемлемых технических характеристик.

Различаются два основных способа приобретения знаний: *настройка* и *адаптация*. Настройка обычно выполняется при первоначальном развертывании ЕЯ-системы на конкретном объекте эксплуатации, а также при значительных изменениях проблемной области, относительно которой ведется общение. Как правило, настройка ЕЯ-системы осуществляется в отсроченном режиме. Состав и объем приобретаемых в процессе настройки знаний, а также требования к уровню профессиональной подготовки персонала, осуществляющего настройку, зависят от особенностей конкретной ЕЯ-системы.

Адаптация заключается в оперативном приобретении знаний ЕЯ-системой в процессе решения задач пользователя. Инициатором адаптации обычно является система. При этом поскольку адаптация выполняется в процессе взаимодействия с пользователем, то никаких специальных требований к его знаниям не предъявляется.

1.4.2. Настройка ЕЯ-систем. В существующих ЕЯ-системах предусматривается настройка по различным параметрам (видам проблемно-ориентированных знаний). К основным из них относятся: знания о языке общения и сущностях проблемной области; знания о пользователе, решаемых им задачах и структуре диалога, направленного на решение этих задач; знания об особенностях прикладной системы, взаимодействие с которой поддерживается ЕЯ-системой. Большинство ЕЯ-систем имеет определенную прикладную ориентацию (общение с БД, диалоговое решение задач, обработка связанного текста). Это позволяет исключить необходимость настройки по некоторым из параметров и улучшить временные и объемные характеристики системы. Так, общение с БД характеризуется относительной устойчивостью решаемых пользователями задач. Поэтому в большинстве ЕЯ-систем для общения с БД предусматривается настройка по первому и третьему параметрам [Grosz et al., 1987]. Для систем обработки связанных текстов характерна жесткая «привязка» к средствам хра-

нения информации, поэтому системы данного класса допускают настройку по первому и (в значительной степени) по второму параметрам. Системы диалогового решения задач в принципе должны иметь возможность настраиваться по всем трем указанным параметрам. Однако в настоящее время большинство из них ориентируется на заранее определенные средства решения задач (в частности, на определенные экспертные системы). Поэтому настройка системы диалогового решения задач, как правило, допускается по первому и второму параметрам.

В зависимости от требований, предъявляемых к уровню профессиональной подготовки лиц, осуществляющих настройку, системы ЕЯ-общения могут быть подразделены на следующие классы: 1) системы, для настройки которых требуется знание лингвистики; 2) системы, настраиваемые администратором базы данных; 3) системы, настраиваемые специалистом в прикладной проблемной области (прикладным специалистом) или непосредственно конечным пользователем. Системы, составляющие последний класс, получили название «переносимых» (transportable) [Grosz et al., 1987].

Необходимость знания лингвистики для настройки системы характерна для ранних ЕЯ-систем (ПОЭТ, МИВОС, LIFER [Hendrix, 1977], SOPHIE и т. п.). Настройка подавляющего большинства современных ЕЯ-систем может выполняться администратором базы данных или прикладным специалистом (т. е. людьми, от которых не требуется знания лингвистики).

Типичным примером ЕЯ-системы, настройка которой осуществляется администратором базы данных, может служить система INTELLECT [Harris, 1977]. Данная система осуществляет поиск и выдачу информации из одного файла БД по запросам пользователя на английском языке. Настройка системы заключается в пополнении (или изменении) ее словаря. Словарь содержит слова двух типов: функциональные — они поставляются вместе с системой и слова, которые вводятся в процессе настройки или адаптации.

К функциональным словам относятся: *что, когда, в, все, какие, сколько, напечатать* и т. п. Семантика этих слов заранее известна системе и не может изменяться при использовании системы в любом из приложений. Фактически функциональные слова являются ключевыми при синтаксическом и семантическом анализе вопросов пользователя.

Слова второго типа отражают специфику конкретной проблемной области, отображаемой в файле БД. К ним относятся: *служащий, фамилия, город, зарплата* и т. п. По существу, они являются внешними именами (и, возможно, значениями) атрибутов файла БД, общение с которым обеспечивает система. Эти слова, а также их синонимы определяются администратором базы данных (исходя из логической структуры файла) и с помощью специального языка вводятся в систему. Поскольку администратор базы данных не может заранее предусмотреть всех слов, соответствующих значениям атрибутов, то они могут вводиться в словарь в процессе адаптации системы (см. ниже) в результате уточняющих поддиалогов, инициируемых системой.

Примерами переносимых систем могут служить такие, как АИСТ. TEAM [Grosz et al., 1987] или IRUS [Bates, 1984]. В отличие от системы INTELLECT они обеспечивают ЕЯ-общение не с одним, а с несколькими связанными файлами БД. При этом настройка допускается не только на язык общения и сущности проблемной области, но и на логическую структуру БД и тип СУБД (в рамках заданного класса СУБД).

Переносимость налагает на разработчиков ЕЯ-системы достаточно жесткие ограничения, которые в полном объеме в существующих системах пока не выполняются. Во-первых, используемый синтаксис должен задаваться общей грамматикой ЕЯ, а не проблемно-ориентированным множеством правил. Это позволяет исключить из процесса настройки модификацию грамматики языка общения. Во-вторых, поскольку семантика языка общения не может зависеть от некоторого проблемно-ориентированного синтаксиса, выбранного для упрощения семантического анализа, то должен быть разработан общий механизм приобретения и присоединения семантики к широкому множеству синтаксических

конструкций. В-третьих, лексика языка общения (словарь) также должна пополняться в процессе настройки. При этом вся необходимая информация специфицируется в терминах, ориентированных на пользователя или прикладного специалиста. И наконец, в-четвертых, процесс доступа к БД должен иметь возможность использовать приобретаемые в процессе настройки знания для перевода того, что было сказано в запросе пользователя, в запрос, соответствующий логической структуре БД. При этом от пользователя и от процесса анализа скрыто, присутствует ли требуемая пользователем информация явно или выводится из информации, имеющейся в БД.

Как правило, в переносимых ЕЯ-системах процесс настройки рассматривается как отдельная задача, которая может решаться либо средствами самой ЕЯ-системы (АИСТ, ТЕАМ), либо с помощью специальной, дополнительной ЕЯ-системы. Последнее имеет место в системе IRUS, для настройки которой применяется специальная ЕЯ-система IRACQ [Moser, 1984]. Блок настройки позволяет «переложить» инициативу и значительную часть ответственности за процесс настройки с лица, производящего настройку (пользователя или прикладного специалиста), на систему. При этом пользователь или прикладной специалист избавляются от заботы о полноте и непротиворечивости вводимой в систему информации, а также от необходимости знания особенностей используемых встраиваемой ЕЯ-системе способов представления проблемно-ориентированных знаний.

Более подробные сведения о настройке ЕЯ-систем по различным параметрам приводятся при рассмотрении конкретных систем (§ 2.3 — ТЕАМ, IRUS), (§ 3.1 — АИСТ).

1.4.3. Адаптация ЕЯ-систем. Основу процесса адаптации составляет оперативное приобретение знаний, недостающих ЕЯ-системе для обеспечения эффективного удовлетворения информационно потребностей конечного пользователя в конкретных ситуациях общения. Необходимость адаптации ЕЯ-систем обусловлена тем, что при настройке системы, как правило, не удается предусмотреть все будущие изменения проблемной области.

По сравнению с настройкой адаптация является более сложным процессом, так как оперативный (т. е. в процессе решения задач пользователя) характер ее выполнения требует в общем случае многоаспектной интерпретации высказываний пользователя (т. е. интерпретации на несколько областей знаний системы (см. § 1.2)). В настоящее время эта задача еще не имеет общего решения, которое может быть рекомендовано для практического применения. В связи с этим возможности адаптации действующих ЕЯ-систем, как правило, весьма ограничены. В большинстве существующих ЕЯ-систем предусматривается адаптация к терминологии, используемой в высказываниях пользователя. Следует также отметить возросший в последнее время интерес к адаптации ЕЯ-систем к уровню квалификации пользователя и решаемым им задачам [Вильштер и др., 1986]. Однако разрабатываемые для решения данной задачи методы пока не вышли из стадии лабораторных исследований.

Рассмотрим адаптацию к терминологии пользователя. Обычно она выполняется в процессе уточняющего поддиалога, инициируемого системой при обнаружении в высказывании пользователя неизвестных ей слов. При этом система использует заранее заготовленный сценарий уточняющего поддиалога. Приведем три типовых сценария, получившие наиболее широкое распространение в современных ЕЯ-системах.

Согласно первому сценарию пользователь «приравнивает» новый термин (или фразу) к структурам, уже известным системе [Hendrix, 1977]. Например: *«Кто преподает дискретную математику?» — Слово преподает неизвестно. — Преподает означает то же, что и ведет занятия. — Дискретную математику преподает доцент Петров.*

В соответствии со вторым сценарием [Martin et al., 1983; Moser, 1984] система с помощью последовательности простых вопросов предлагает пользователю выработать необходимое определение, в результате чего она получает как синтаксическую, так и семантическую информацию о новом термине. Фак-

тически в данном случае управление передается блоку приобретения знаний, используемому при настройке ЕЯ-системы.

Третий сценарий основан на предположении, что неизвестный термин представляет собой значение, имеющееся в базе данных. Такое предположение оправдано, если ЕЯ-система сопряжена с большой, быстро изменяющейся базой данных. В этом случае нецелесообразно включать в словарь все значения базы данных, а смысл неизвестного слова может рассматриваться как имя поля (атрибута) базы данных, чьим значением служит данное слово. Примером системы, использующей этот сценарий (вместе с первым из рассмотренных сценариев), может служить ЕЯ-система INTELLECT. Приведем пример уточняющего диалога, использующего данную стратегию. В примере действия пользователя обозначаются П, а системы — С.

П: Сообщите имена бухгалтеров в Пафкипси.

С: Слово «Пафкипси» системе неизвестно.

Если вы предполагаете найти это слово в базе данных, нажмите клавиш «Возврат». В противном случае исправьте его написание или введите синоним.

П: <нажимает клавиш «Возврат»>

С: В каком поле оно должно появиться?

П: Город.

С: Напечатать фамилии всех служащих с город-Пафкипси и должность-бухгалтер.

Число записей в отчете=6: Потвин Д. Р., Скотт Г. М., ...

В заключении отметим, что для действующих ЕЯ-систем характерно использование не какого-либо одного, а сразу нескольких из рассмотренных сценариев. Поэтому при обнаружении незнакомого термина система предоставляет пользователю возможность выбора одного из имеющихся у нее сценариев (см. предыдущий пример).

Глава 2.

Состояние развития современных ЕЯ-систем

А. Б. Преображенский

2.1. Основные классы ЕЯ-систем

В § 1.2 были определены и в общем виде рассмотрены основные функциональные компоненты ЕЯ-систем: ведение диалога, понимание высказываний и генерация высказываний. В зависимости от назначения прикладных систем, в состав которых входят конкретные реализации ЕЯ-систем, задачи, решаемые отдельными функциональными компонентами (как по постановке, так и по методам решения), могут в значительной степени варьироваться. Исходя из этого, а также учитывая историю развития ЕЯ-систем, будем различать следующие основные классы: 1) интеллектуальные вопрос-ответные системы; 2) системы общения с базами данных; 3) диалоговые системы решения задач; 4) системы обработки связанных текстов.

Исторически ЕЯ-системы происходят от информационно-поисковых систем, с одной стороны, и систем машинного перевода, с другой [Winograd, 1983]. Поэтому на начальном этапе ЕЯ-системы представляли собой макеты информационно-поисковых систем, демонстрирующие принципиальную возможность ввода данных (фактов) и обработки запросов на естественном языке. Такие системы часто назывались интеллектуальными вопрос-ответными системами. Название можно, по-видимому, объяснить стремлением их разработчиков подчеркнуть, что в отличие от обычных информационно-поисковых систем и си-

стем машинного перевода того времени в данных системах широко используются концепции, выработанные в исследованиях по ИИ.

Основное внимание при разработке интеллектуальных вопрос-ответных систем уделялось не столько возможностям их практического использования в реальных задачах, сколько развитию моделей и методов, позволяющих осуществлять перевод ЕЯ-высказываний, относящихся к узким и заранее фиксированным проблемным областям, в формальное представление, а также обратный перевод. Накопленный опыт разработки интеллектуальных вопрос-ответных систем позволил, с одной стороны, углубить понимание процесса ЕЯ-общения и, следовательно, поставить новые проблемы (в том числе и специфичные для общения в различных классах проблемных областей), требующие дальнейшей проработки, а с другой — оценить перспективы практического применения ЕЯ-систем.

Первые предпосылки для практического использования ЕЯ-систем создало появление баз данных (БД). В связи с этим возникла проблема обеспечения доступа к информации, хранящейся в БД, широкому классу неподготовленных конечных пользователей, к которым относят специалистов в той или иной предметной области, как правило, не обладающих знаниями о логической структуре БД, о системе представления информации в БД и не умеющих пользоваться формализованными языками запросов. Для решения этой проблемы стали создаваться системы общения с базами данных, основная задача которых (в простейшем случае) заключается в выполнении перевода запросов неподготовленных конечных пользователей с ЕЯ на формализованные языки запросов к БД.

Диалоговые системы решения задачи в отличие от систем общения с БД берут на себя не только функции ЕЯ-доступа к БД, но и функции интеллектуального монитора, обеспечивающего решение заранее определенных классов задач (например, планирование путешествий, боевых операций, составление контрактов и т. п.). В этом случае разбиение задач на подзадачи и распределение ролей между участниками, т. е. определение, кто из участников (пользователь или система) решает ту или иную подзадачу, осуществляется не пользователем (как в случае применения систем общения с БД), а диалоговой системой. Решение подзадач «порученных» системе, может осуществляться как на основе использования собственных знаний и механизмов вывода, так и в результате обращения к прикладным программам и пакетам, не входящим в состав ЕЯ-системы. Основным направлением практического использования ЕЯ-систем данного класса является реализация ЕЯ-общения с экспертными системами.

Возникновение последнего класса ЕЯ-систем — систем обработки связанных текстов обусловлено возрастанием объема хранимой в ЭВМ текстовой информации (газетные статьи, сообщения о различных событиях, патенты, авторские свидетельства и т. п.), и необходимостью извлечения из нее разнообразных сведений (например, о структуре некоторых объектов, о действующих лицах некоторых событий, о мотивах их поступков и т. д.).

Каждый из классов ЕЯ-систем обладает специфическими особенностями, которые хорошо заметны при рассмотрении характера задач, решаемых основными функциональными компонентами этих систем (табл. 2.1).

Приведенная выше классификация ЕЯ-систем охватывает лишь функционально полные системы, т. е. такие, в которых представлены все основные функциональные компоненты. Однако помимо функционально полных систем ведется интенсивная разработка систем, которые можно назвать фрагментарными. Цель их создания — исследование или реализация новых методов решения достаточно узких задач (например, таких, как анализ, интерпретация, определение целей пользователя и т. п.). Благодаря модульности структуры ЕЯ-систем и, как правило, универсальному (т. е. не зависящему от специфики прикладных областей) характеру языка внутреннего представления, фрагментарные системы могут успешно использоваться в качестве отдельных функциональных блоков, встраиваемых (хотя бы на логическом уровне) в различные функционально полные ЕЯ-системы.

В табл. 2.2 приведены наиболее общие сведения о существующих ЕЯ-системах, почерпнутые из материалов международных и национальных конференций

Сравнительная характеристика основных классов ЕЯ-систем

Компоненты ЕЯ-системы	Интеллектуальные вопрос-ответные системы	Системы общения с базами данных	Диалоговые системы решения задач	Системы обработки связанных текстов
Ведение диалога	Жесткий диалог с инициативой у пользователя	Жесткий или альтернативный диалог. Инициатива в основном принадлежит пользователю. Перехваты инициативы системой для предупреждения или устранения некоторых видов логических неудач	Гибкий диалог. Инициатива в основном принадлежит системе. Разнообразные способы перехвата инициативы	Жесткий диалог с инициативой у пользователя
Понимание высказываний	Отдельные правильные ЕЯ-предложения с лексическими, синтаксическими и семантическими ограничениями. Интерпретация осуществляется на проблемную область	ЕЯ-предложения, которые могут содержать некорректности и ссылки на предыдущие запросы. Интерпретация на проблемную область или область языка общения	ЕЯ-предложения, которые могут содержать некорректности и ссылки на предыдущие высказывания. Интерпретация на проблемную область, область языка общения, область системы, область пользователя и область дискурса	Связный текст в виде последовательности ЕЯ-предложений, описывающих события, сложные объекты, явления и т. п.
Генерация высказываний	Прямые ответы в виде ЕЯ-предложений с лексическими, синтаксическими и семантическими ограничениями. Стандартные сообщения о неудачах	Прямые ответы, получаемые с помощью средств генерации отчетов СУБД. Стандартные сообщения для предупреждения или устранения неудач. Перефразирование запросов на ЕЯ	Прямые и косвенные ответы в виде ЕЯ-предложений, генерируемых с учетом целей и намерений участников общения	Связный текст в виде последовательности ЕЯ-предложений

Основные сведения о ЕЯ-системах

Название системы (страна)	Назначение	Язык программирования, тип ЭВМ	Источник
ADVISOR (США)	Общение с экспертной системой на английском языке		McKeown. et al., 1985
ARGOT (США)	Моделирование процесса ведения диалога как процесса достижения целей		Allen et al., 1982
ARTUR (США)	Понимание связанного текста на английском языке на основе построения планов действующих лиц		Grander, 1980
ATLAST (США)	Интерпретация связанного текста на английском языке	UCI-LISP, DECSYSTEM-20	Eiselt, 1985
ВАОВAB-2 (США)	Понимание медицинских заключений на структурированном, ограниченном английском языке	INTERLISP, DEC KA-10	Bonnet, 1979
CASPAR (США)	Понимание запросов на английском языке к базе данных		Hayes et al., 1981a
CO-OP (США)	Понимание запросов на английском языке к изменяющейся базе данных		Kaplan et al., 1979
CYRUS (США)	Интерпретация связанного текста на английском языке	DECSYSTEM-20	Shank et al., 1977
DIAMOND (США)	Понимание отдельных высказываний на английском языке	INTERLISP, PDP-10	Paxton, 1978
DONAU (Италия)	Анализ отдельных высказываний на итальянском языке, адаптируемый к проблемной области	LISP, UNIVAC-1110	Гвида и др., 1981
FAS-80 (ГДР)	Интеллектуальная вопрос-ответная система. Язык общения — немецкий	ЛИСП, ЕС-1055	Helbig et al., 1983
FAUSTUS (США)	Интерпретация связанного текста на английском языке		Norvig, 1983
FIDO (Италия)	Анализатор для итальянского языка	FRANZLISP, VAX 11/730	Lesmo et al., 1985

Название системы (страна)	Назначение	Язык программирования, тип ЭВМ	Источник
Flex-P (США)	Понимание высказываний на английском языке	DECSYSTEM-20	Hayes et al., 1981б
FRUMP (США)	Понимание связанного текста на английском языке		De Jong, 1979
GLS (США)	Диалоговая система решения задач с английским языком общения		Bobrow et al., 1977а
HAM-ANS (ФРГ)	Многоцелевая система общения на немецком языке с настройкой на прикладные системы: БД, экспертная система, система машинного зрения	UCI-LISP/FUZZY. DEC-1070	Hoerpner et al., 1983
HAM-PRM (ФРГ)	Диалоговое решение задач информационного обслуживания на немецком языке		Hahn et al., 1980
NWIM (США)	Понимание высказываний на английском языке	INTERLISP	Woods et al., 1976
INTELECT (США)	Система общения с БД на английском языке		Harris, 1977
IPP (США)	Понимание связанного текста на английском языке		Шенк и др., 1983
IRUS (США)	Общение с БД на английском языке с настройкой на базу данных и тип СУБД	FRL	Bates, 1984
JETS (США)	Общение с изменяющейся БД на английском языке		Finin et al., 1979
KAMP (США)	Ведение диалога на основе использования механизма планирования	Ассемблер, ЕС ЭВМ INTERLISP, PDP KL-10	Appelt, 1982
KING-KONG (США)	Анализ высказываний на английском языке		Bayer et al., 1985
KODAS (ЧССР)	Общение с БД на чешском языке		Hajic, 1983
LIFER (США)	Общение с БД на английском языке		Hendrix, 1977
LUNAR (США)	Интеллектуальная вопрос-ответная система. Язык общения — английский		Woods et al., 1978

Название системы (страна)	Назначение	Язык программирования, тип ЭВМ	Источник
MARGE (США)	Понимание связанного текста на английском языке	MLISP, PDP-10	Шенк, 1980
MENTAT (Франция)	Общение с БД на французском языке. Преду- смотрена настройка на проблемную область		Ogonowski, 1987
MULTIPAR (США)	Понимание команд к операционной системе, вы- раженных на английском языке		Minton et al., 1985
MUMBLE (США)	Генерация высказываний на английском языке		McDonald et al., 1985
RAM (США)	Понимание связанных текстов на английском языке	INTERLISP, Siemens 7748	Wilensky, 1979
RAMELA (США)	Интерпретация запросов о содержании связанных текстов на английском языке		Luria, 1982
PARNAX (Италия)	Общение с БД на итальянском языке		Comino et al., 1983
PARSIFAL (США)	Анализ высказываний на английском языке		Shipman et al., 1979
PENMAN (США)	Генерация связанного текста на английском языке в соответствии с заданной целью	LISP	Mann, 1983
PHRAN (США)	Анализ высказываний на английском, испанском и китайском языках	FRANZLISP, VAX 11/780	Wilensky, 1981a
PHRED (США)	Генерация высказываний на английском и испан- ском языках		Jacobs, 1983
PLANIS (США)	Общение с БД на английском языке		Waltz, 1978
PLAS (Индия)	Моделирование процесса обучения ЕЯ-системы по аналогиям		Sembugamoorthy, 1979
Q-TRANS (США)	Генерация перефраз на английском языке	UCI-LISP, DEC-10	Mueckstein, 1983
RESEARCHER (США)	Обработка связанных текстов на английском языке		Lebowitz, 1985
SPRUS (США)	Анализ высказываний на английском языке		Bobrow, 1980
		DEC KL-10	

Название системы (страна)	Назначение	Язык программирования, тип ЭВМ	Источник
SAPHIR+RESEDA (Франция, Италия)	Общение с БД с настройкой на проблемную область и тип СУБД		Eusenat et al., 1985
SDRL (ССР)	Общение на румынском языке с экспертной системой	LISP	Tufis, 1982
SHDRLU (США)	Понимание высказываний на английском языке	PROGRAMER, PLANNER, LISP	Виноград, 1976
SNUKA (США)	Общение с экспертной системой на английском языке		Pazzani, 1983
SOPHIE (США)	Диалоговое решение задач на английском языке	LISP	Brown et al. 19756
SUSAN (ГДР)	Синтаксический анализ запросов к БД на немецком языке	ЛИСП ЕС	Koch et al., 1982
SUSY (Италия)	Генерация рефераторов на итальянском языке по связным текстам		Fum et al., 1985
TEAM (США)	Общение с БД на английском языке с настройкой на проблемную область и тип СУБД	INTERLISP, DEC-2060, Symbolics, LM-2	Grosz et al., 1987
TELEGRAM (США)	Генерация высказываний на английском языке с учетом коммуникативных целей		Appelt, 1983
TEXT (США)	Генерация высказываний на английском языке с учетом целей и планов участников общения		McKeown, 1983
THEMES (США)	Общение с БД на английском языке. Предусмотрена настройка на проблемную область и тип БД	DEC VAX, IBM PC/AT	Expert Systems User, 1987, Vol. 2, № 10
TQA (США)	Общение с БД на английском языке		Petric, 1981
TRACK (США)	Моделирование механизма формирования целей говорящего при диалоговом решении задач		Carberry, 1983
TULIPS (СССР)	Понимание высказываний на русском языке	ПЛЭННЭР, Лисл, БЭСМ-6	Мальковский, 1985
UC (США)	Общение на английском языке с консультантом по ОС UNIX	FRANZLISP, VAX 11/780	Wilensky, 1982

Название системы (страна)	Назначение	Язык программирования, тип ЭВМ	Источник
VOX (США)	Анализ и интерпретация на область языка общения высказываний на английском языке	FRANZLISP, VAX 11/780	Carbonell et al., 1981
XCALIBUR (США)	Общение с экспертными системами на английском языке		Carbonell et al., 1983
АДАЛИТ (СССР)	Понимание вопросов на русском языке. Преду- смотрена настройка на проблемную область	Лисп, ЕС ЭВМ	Васильев и др., 1983
АДС (СССР)	Диалоговая система решения задач, настраи- ваемая на проблемную область	Лисп, СМ-4	Брябрин, 1981
АИСТ (СССР)	Общение с БД на русском языке; диалоговое ре- шение информационно-поисковых задач	Ассемблер, ЕС ЭВМ	Дракин и др., 1988
ДИЛОС (СССР)	Диалоговое решение задач на русском языке	Лисп, БЭСМ-6	Брябрин и др., 1983
ДИС 1.0 (СССР)	Общение с БД на русском языке. Построение информационно-поисковых систем. Обнаружение орфографических ошибок	Си, ЕС-1841	Кирсанов и др., 1989
ПИРС 2.0 (СССР)	Построение систем обработки данных с ЕЯ-интер- фейсом, средством деловой и иллюстративной графики, электронной таблицей (развитие систе- мы ДИС 1.0)	То же	То же
ДИСАР (СССР)	Диалоговая система решения задач автоматизи- рованного проектирования. Общение на ограни- ченном русском языке	CDL, ЕС-1010	Рыбина, 1982
ДИСПУТ (СССР)	Общение с БД на русском языке	Фортран, Ассемблер ЕС ЭВМ	Брябрин и др., 1983
ЗАПСИБ-10 (СССР)	Общение с БД на русском языке	ПЛ/1, ЕС ЭВМ	Ангелова и др., 1984
ИПС (ПНР)	Общение с БД на польском языке	Пролог	Szrakowicz et al., 1982
ЛИНГВИСТ (СССР)	Общение с БД на русском языке	Си, ЕС-1841	Попов и др., 1988
МИВОС (СССР)	Интеллектуальная вопрос-ответная система. Язык общения — русский. Предусмотрена настройка на проблемную область	РЕФАЛ, БЭСМ-6	Преображенский, 1982
ПОЭТ (СССР)	Интеллектуальная вопрос-ответная система с рус- ским языком общения	ПЛ/1, ЕС ЭВМ	Попов, 1982
ТАСС (СССР)	Система обработки связанного текста на русском языке		Пранц, 1983

по искусственному интеллекту, а также публикаций обзорного характера. Графа «Источник» содержит ссылки на публикации, приведенные в библиографии. В графе «Назначение» указывается принадлежность ЕЯ-системы к одному из рассмотренных выше классов, а для фрагментарных систем — реализуемые функции.

Размеры настоящего справочника не позволяют дать подробное описание всех систем, включенных в табл. 2.2. По-видимому, в этом и нет необходимости, так как многие из них (разработанные в период 1976—1982 гг.) детально рассматривались в ряде широко известных работ (см., например, [Попов, 1982; Представление, 1984; Уилкс, 1983; Шенк, 1980]). Тем не менее, учитывая справочный характер настоящего издания, мы стремились включить в данную таблицу максимально возможное количество известных нам систем. При этом имело в виду, что для оценки уровня развития ЕЯ-систем будут использоваться системы, сведения о которых появились в последнее время, а «старые» системы, многие из которых являются как бы вехами на пути развития ЕЯ-систем, будут использоваться при необходимости обращения к ретроспективе.

2.2. Интеллектуальные вопрос-ответные системы

Как уже говорилось выше, при разработке интеллектуальных вопрос-ответных систем основное внимание уделялось языковому аспекту, т. е. максимальному приближению языка общения к литературному естественному языку. Наиболее значительной из отечественных систем данного класса является система ПОЭТ, во многом определившая применяемые в последующих системах методы анализа и генерации высказываний на русском языке.

Система ПОЭТ (мы здесь имеем в виду ее первую версию [Попов, 1982]) воспринимает вопросительные предложения русского языка с практически несущественными ограничениями на допустимые синтаксические конструкции и пунктуацию. Типичными примерами запросов, допускаемых системой ПОЭТ, могут служить «Сколько каменного угля перевезено железнодорожным транспортом в 1978 году?» или «Каков удельный вес перевозок железнодорожным транспортом в общем объеме перевозок всеми видами транспорта в 1975 году?»

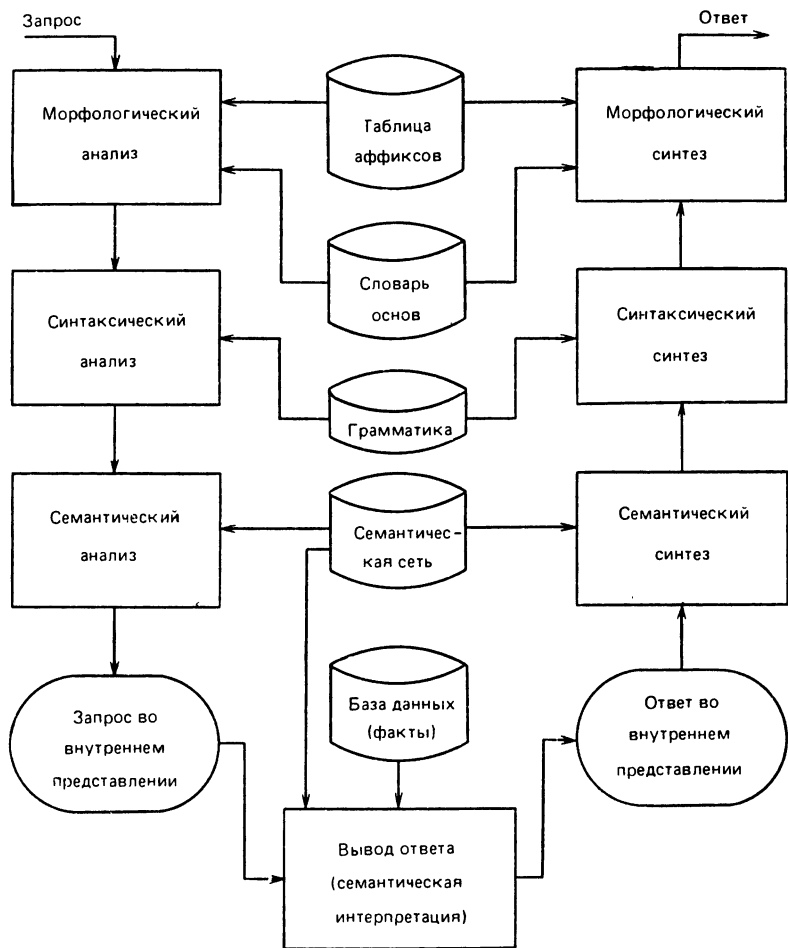
Процесс понимания входных высказываний осуществляется в системе ПОЭТ по полной схеме: морфологический анализ, синтаксический анализ, семантический анализ и семантическая интерпретация (рис. 2.1). При этом последние три этапа выполняются в общем случае параллельно, за счет чего достигается коррекция неверных путей анализа и в конечном счете сокращается время обработки запросов.

Все знания о языке общения разделяются в системе ПОЭТ на лингвистические и проблемные. Первые хранятся в различных зонах словаря (морфологической и синтактико-семантической), а вторые — в семантической сети. При этом в системе различаются абстрактная (описывающая общие понятия и категории) и конкретная (описывающая конкретные сущности) семантические сети.

Синтаксический анализ в системе ПОЭТ направляется значениями базовых событий, обнаруживаемых в анализируемом высказывании. Описания базовых событий представлены в словаре системы в виде моделей управления. Выделение участников событий и определение выполняемых ими ролей осуществляются на основе метода фильтров. При этом активно используется как грамматическая, так и синтактико-семантическая информация.

На этапе семантического анализа синтаксическая структура входного высказывания, представленная в виде дерева зависимостей, преобразуется в семантический граф, состоящий из вершин-понятий, связанных друг с другом через вершины-события и характеристики. Каждая вершина семантического графа определяется каноническим представлением, а дуги имеют глубинный смысл. Вся числовая и параметрическая информация выносится из графа в дополнительные таблицы. Там же указываются и временные соотношения между событиями.

На этапе интерпретации семантический граф запроса сопоставляется с семан-



тической сетью. В результате происходит вычленение контекста, имеющего отношение к запросу, получение содержательной информации из конкретной сети, формирование обращений к базе данных за числовой информацией и получение способа обработки этой информации (суммирование, вычисление процента и т. п.). Семантический граф ответа вырабатывается на базе графа запроса внесением в него смысловой информации, полученной на этапе интерпретации.

генерация поверхностной структуры ответа. Генерация полного ответа, например «В 1978 году железнодорожный транспорт перевез NNN млн. тонн каменного угля», позволяет пользователю убедиться в правильности понимания системой заданного вопроса.

С помощью первых вопрос-ответных систем была показана принципиальная возможность получения ответов на ЕЯ-вопросы, относящиеся к ограниченным проблемным областям. Так, система ПОЭТ могла отвечать на вопросы о перевозках различных народно-хозяйственных грузов, ДИСПУТ — об обслуживании контейнерных перевозок к морскому порту, LUNAR — о свойствах образцов лунных пород, LIFER — о дислокации и характеристиках судов военно-морских сил и т. п. Для большинства интеллектуальных вопрос-ответных систем была характерна жесткая структура диалога, при которой каждое высказывание пользователя воспринималось как очередной запрос (который, как правило, не мог быть связан с предыдущим). Система играла пассивную роль — она могла лишь отвечать на запросы и выдавать сообщения о неудачах, когда очередной запрос по каким-либо причинам не мог быть проанализирован или обработан. Обработка высказываний сводилась в большинстве случаев к вызову (в соответствии с типом запроса) одной из имеющихся в распоряжении системы специализированных программ и передаче ей в виде параметров условий поиска информации в БД, имен сущностей, значения которых должны быть обработаны или выданы в качестве ответа, и т. п.

Первые эксперименты с интеллектуально вопрос-ответными системами показали, что, несмотря на возможность понимания запросов на ЕЯ, данные системы налагают достаточно жесткие (и в общем случае трудновыполнимые) ограничения на процесс общения. Эти ограничения сводились, по крайней мере, к следующим. Во-первых, информационная потребность пользователя (как данных, так и процедур их обработки) должна была соответствовать возможностям системы, заложенным при ее разработке. При этом, поскольку любой запрос интерпретировался на проблемную область, пользователь не мог задавать вопросы о возможностях системы. Во-вторых, представления пользователя о языке общения и проблемной области должны были точно совпадать со знаниями о языке и проблемной области, имеющимися у системы и отражающими соответствующие представления разработчиков. В-третьих, жесткая структура диалога с инициативой у пользователя требовала полного выражения информационной потребности в рамках одного запроса, что, вообще говоря, противоречит стилю естественно-языкового общения, приводит к увеличению длины запроса и, как следствие, — к увеличению вероятности появления ошибок. И наконец, в-четвертых, при выражении запросов позволялось использовать только «правильные» предложения, т. е. такие, которые система могла однозначно понять и верно обработать.

Эти ограничения стимулировали дальнейшие исследования, направленные в первую очередь на повышение гибкости процесса общения. Вместе с тем, как уже говорилось ранее, потребности практического использования ЕЯ-систем привели к формированию таких классов системы, как системы общения с БД, диалоговые системы решения задач и системы обработки связанных текстов.

2.3. Системы общения с базами данных

Концепция базы данных лежит в основе подавляющего большинства современных систем обработки данных. Для обеспечения взаимодействия с конечными пользователями системы управления базами данных (СУБД) предоставляют специальные формализованные языки. Однако, как правило, эти языки ориентированы на пользователей, обладающих специальными знаниями. В частности, они должны знать основные приемы программирования, синтаксис языка запросов, логическую структуру БД, термины, используемые в БД для обозначения сущностей предметной области, и т. п. ЕЯ-системы рассматриваемого класса предназначены для использования в качестве посредника (интерфейса) между неподго-

P1: сколько сотрудников 45 отд не гроживают в Кембридже или Бостоне
 C1: 19
 P2: перечислите их имена
 C2: фамилия имя
 Бейтс Мэдэлич
 Бобров Роберт

 P3: есть ли у кого-нибудь из них публикации по синтаксису
 C3: да, у 1 из 19
 P4: у кого
 C4: фамилия имя
 Бейтс Мэдэлич
 P5: кто из 45 от1 живет в Арлингтоне?
 C5: отъезд?
 P6: да
 C6: фамилия имя
 Ларкин Кэтти

 P7: 41?
 C7: фамилия имя
 Кэглайн Альберт

 P8: журналы, финансируемые IEEE?
 C8: IEEE SPECTRUM
 IEEE TRANS.SYST.MAN.CYBER
 P9: приведите 10 наиболее свежих статей о синтаксисе
 C9: извините, имеется только 4 таких статьи
 PLANNING UNITS AND SYNTAX IN SENTENCE PRODUCTION
 I.....
 P10: действительно ли Смит написал все статьи про медведей коала?
 C10: извините, в базе данных отсутствуют подобные библиографические сведения

Рис. 2.2. Пример диалога с системой общения с БД

товленными конечными пользователями (т. е. специалистами в прикладных проблемных областях, не обладающими указанными выше знаниями) и БД. Другими словами, они должны позволять получать информацию, хранящуюся в БД, по запросам, сформулированным на ЕЯ.

На рис. 2.2 приведен фрагмент диалога [Bates, 1984], демонстрирующий типичные возможности ЕЯ-систем данного класса. Данный пример показывает, что общение ведется в форме связного диалога, т. е. ответы на вопросы пользователя выдаются с учетом его предыдущих вопросов и/или предыдущих ответов системы (C2, C3, C4). Инициатива в диалоге в основном принадлежит пользователю. Перехват инициативы допускается лишь для уточнения незнакомых системе слов и исправления орфографических ошибок (C5). Язык общения является подмножеством естественного языка, семантически ограниченным предметной областью, отображаемой в БД. В высказываниях пользователя допускаются многие общепринятые синтаксические конструкции. Кроме того, допускаются определенные неправильности: орфографические ошибки (P5), пропуск слов (P8), ошибки в пунктуации (P1, P9), неправильное употребление строчных и прописных букв (P1, P9) и ряд типичных диалоговых конструкций: эллипсис (P7) и анафорические ссылки (P2, P3, P4). Ответы ЕЯ-системы строятся таким образом, чтобы обеспечить, насколько это возможно, «дружелюбность» общения. В частности, сообщается о невыполнении пресуппозиций, содержащихся в запро-

се (C10), ответы выдаются избыточными (C3), что позволяет в какой-то степени учесть дальнейшее развитие информационной потребности пользователя.

С точки зрения возможностей практического использования системы общения с БД значительно опережают остальные классы ЕЯ-систем. Существует несколько коммерческих и промышленных ЕЯ-систем общения с БД (см., например, [Дракин и др., 1988; Bates et al., 1984]), которые успешно применяются на практике. Промышленная эксплуатация ЕЯ-систем является мощным источником дополнительных требований, стимулирующих дальнейшее развитие этих систем, которое в первую очередь находит отражение в экспериментальных системах (демонстрационных и исследовательских прототипах).

Ниже будут рассмотрены три ЕЯ-системы общения с БД: PARNAX [Comino et al., 1983], TEAM [Grosz et al., 1987; Martin et al., 1983], IRUS [Bates, 1984], в которых, на наш взгляд, в наибольшей степени учитываются требования и условия промышленной эксплуатации (в частности, обеспечение общения с коммерческими БД). Выбор нескольких систем обусловлен, с одной стороны, разнообразием методов, используемых для решения одних и тех же задач, а с другой — отсутствием полной информации о всех компонентах этих систем.

Система PARNAX. Данная система осуществляет перевод запросов, сформулированных на итальянском языке, в программы на языке NATURAL, который является формализованным языком запросов СУБД ADABAS. В общей структуре системы PARNAX (рис. 2.3) могут быть выделены два основных блока: макроанализатор и микроанализатор.

Макроанализатор выполняет разбиение и нормализацию сложных предложений на простые фрагменты. Например, запрос «Сколько служащих старше 30 лет в отделе информатики и каковы их специальности?» будет разбит на два фрагмента: «Сколько служащих старше 30 лет в отделе информатики?» и «Каковы специальности служащих старше 30 лет из отдела информатики?»

Микроанализатор осуществляет разбор полученных фрагментов и построение для них формальных представлений в виде поддеревьев, выраженных на языке METANATURAL. Данный язык является промежуточным, обеспечивающим независимость процесса понимания входных высказываний от деталей логической структуры (схемы) базы данных. Полученные поддеревья передаются макроанализатору, который строит окончательное представление запроса на языке NATURAL.

На этапе макроанализа используется множество так называемых структурных правил вида <образец, действие по разбиению и нормализации, действие по объединению>. Каждое правило можно применять как при разбиении входного высказывания на отдельные фрагменты, так и при сборке окончательного представления на языке METANATURAL. Макроанализатор включает также блок перефразирования и ведения диалога, который поддерживает диалоговое взаимодействие с пользователем и генерирует перефразы входных запросов.

Процесс микроанализа распадается на три этапа: 1) лексический анализ; 2) семантико-синтаксический анализ; 3) семантическая интерпретация. Лексический анализатор представляет собой недетерминированный конечный распознаватель, использующий лексическую часть словаря для идентификации слов и словосочетаний, образующих фрагменты входных запросов, и определения их грамматических категорий.

На втором этапе выполняются два параллельных процесса: синтаксический анализ, который базируется на грамматике зависимостей, и семантический анализ. Базу знаний, используемую на данном этапе, составляют: правила грамматики зависимостей; семантические правила; семантическая часть словаря.

В результате выполнения второго этапа фрагменту входного запроса сопоставляется множество модифицированных деревьев зависимостей, из которых удалены семантически бессмысленные компоненты и в которые добавлены семантические переменные, соответствующие основным типам информации, выражаемой анализируемыми фрагментами, и фактически являющиеся отсылками к процедурам (семантическим модулям), выполняемым на этапе интерпретации.

Третий этап — семантическая интерпретация — представляет собой детерми-

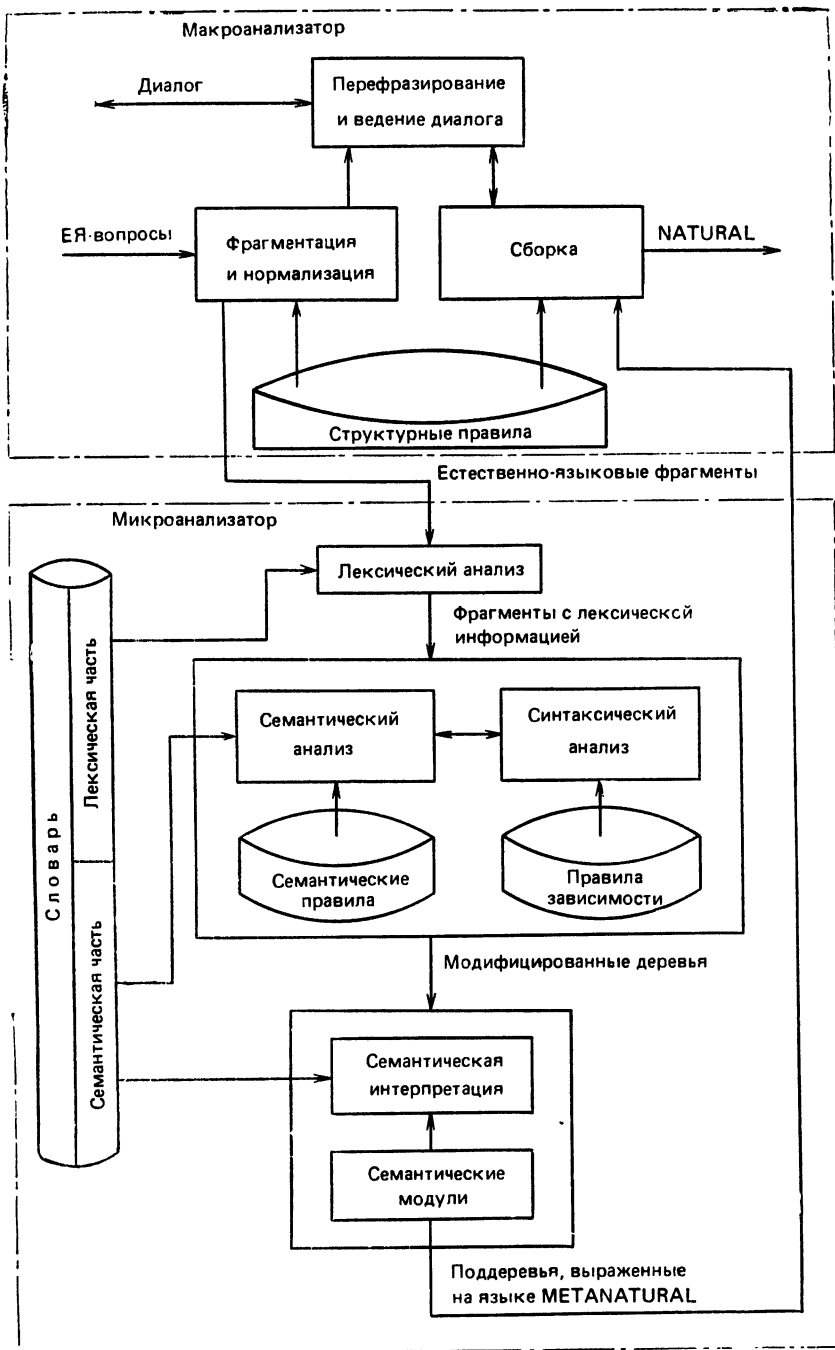


Рис. 2.3. Структурная схема анализатора системы PARNAX

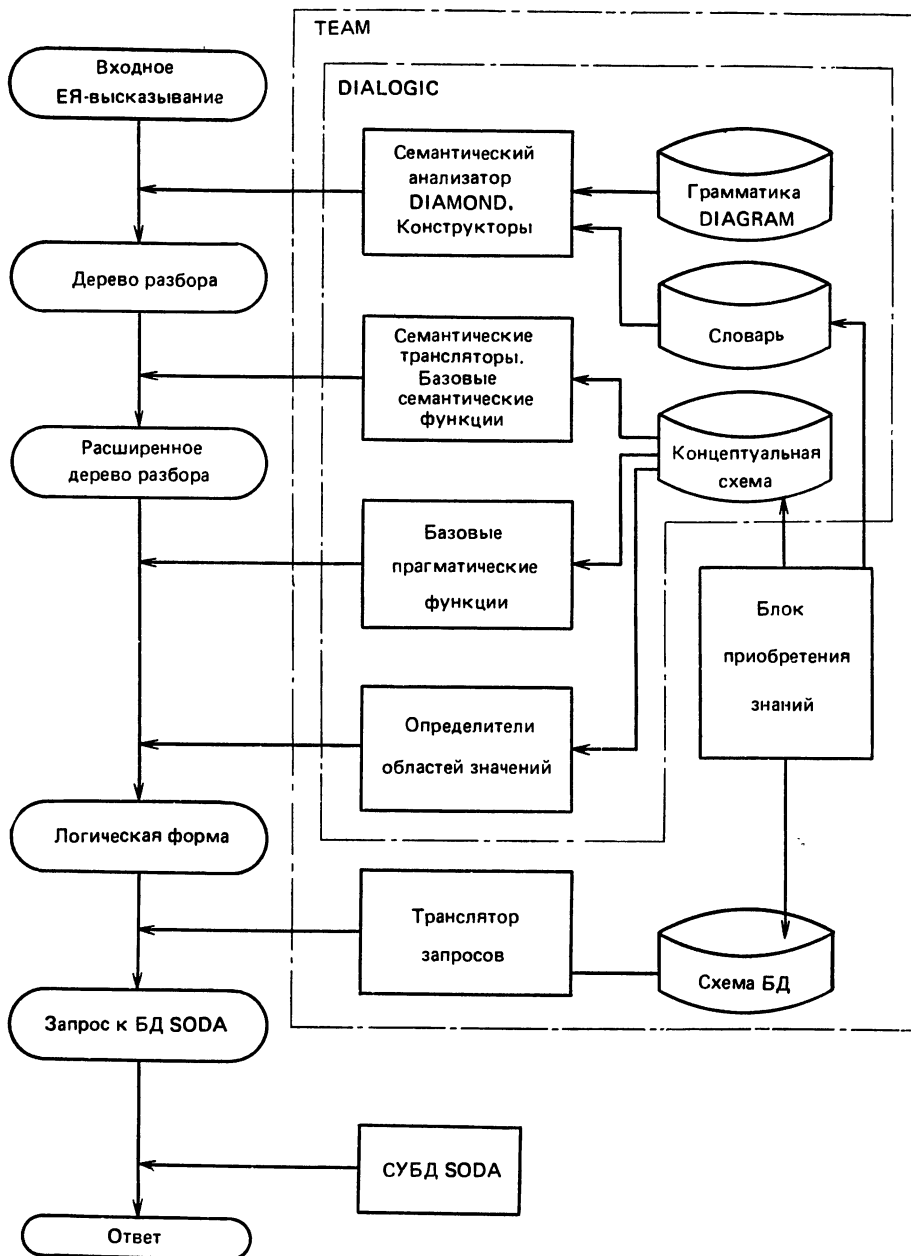


Рис. 2.4. Структурная схема системы TEAM

нированный процесс построения выходных поддеревьев на языке METANATURAL. На данном этапе применяется восходящая стратегия вызова семантических модулей, на которые имеются ссылки в модифицированных деревьях, и присвоение соответствующих значений семантическим переменным.

Система TEAM. Данная система служит примером настраиваемой ЕЯ-системы, обеспечивающей взаимодействие на английском языке с реляционными базами данных. Общая структура TEAM показана на рис. 2.4. Ядром системы TEAM является фрагментарная система DIALOGIC [Grosz et al., 1982], которая осуществляет преобразование входных английских предложений в логическую форму, представляющую их буквальное содержание. Для этого используются два вида информации, полученной в процессе настройки: словарь в виде таблицы слов английского языка с их определениями; концептуальная схема — множество определений (в виде фреймов) предикатов, которые могут появляться в запросах. (Предполагается, что проблемно-независимая информация, например, о местоимениях и предлогах в этих структурах данных дается в виде базовых понятий.)

Работа DIALOGIC состоит из нескольких фаз, показанных на рис. 2.4. На первой фазе выполняется синтаксический анализ входных высказываний. Для этого используется анализатор DIAMOND [Paxton, 1978], управляемый грамматикой DIAGRAM [Robinson, 1982]. Грамматика представляет собой множество контекстно-свободных правил, снабженных специальными дополнениями (конструкторами), позволяющими преодолевать контекстно-зависимые синтаксические ограничения и приписывать разборам и их фрагментам веса (оценки), в основе которых лежит априорное правдоподобие отдельных синтаксических конструкций. Вычисление оценок синхронизировано с процессом восходящего синтаксического анализа. Полученные в результате синтаксического анализа оценки альтернативных разборов позволяют выбрать «наилучший» и передать его в первую очередь на семантический анализ.

Семантический анализ выполняется с помощью так называемых семантических трансляторов (СТ) и базовых семантических функций (БСФ). СТ проблемно-независимы, однако их действия в значительной степени обусловлены данными из концептуальной схемы, которые были получены при настройке системы. СТ как бы проецирует надежные структуры предикатов на синтаксические структуры анализируемых высказываний. БСФ, вызываемые СТ в соответствии с типами аргументов предикатов, расширяют деревья разборов указателями именных групп, предикатов и дополняют их квантифицированными переменными. С помощью БСФ разрешаются некоторые виды неоднозначностей, а также определяются и отбрасываются семантически неправильные разборы.

На фазе прагматического анализа осуществляется два преобразования структуры, полученной в результате синтаксического анализа. Во-первых, разрешаются остающиеся неоднозначности, вызванные неопределенными предикатами (например, такими, как предлог *of*). Для этого используются базовые прагматические функции, которые заменяют неопределенные предикаты на однозначные предикаты, выведенные исходя из свойств типов их аргументов в высказывании. Во-вторых, на данной фазе устанавливаются области действия кванторов, отрицаний и границы суперлативных конструкций.

Результатом анализа является логическая форма, в общем случае представляющая собой высказывание на языке исчисления предикатов высших порядков. В качестве примера на рис. 2.5 представлена (в виде дерева) логическая форма для классического запроса: «What employees earn more than their manager's salary?» (Кто из служащих зарабатывает больше, чем их управляющий?). На рис. 2.5 строчными буквами обозначены следующие переменные: *e* — служащий, *m* — управляющий, *s1* — заработок управляющего, *s2* — заработок служащего.

Получением логической формы завершается работа компонента DIALOGIC. Формирование запроса к БД осуществляется транслятором запросов. Решаемые им задачи сводятся к следующим: 1) замена кванторов на машинно-ориентированные операторы; 2) упрощение логической формы путем расширения определений предикатов, соответствующих виртуальным отношениям; 3) удаление

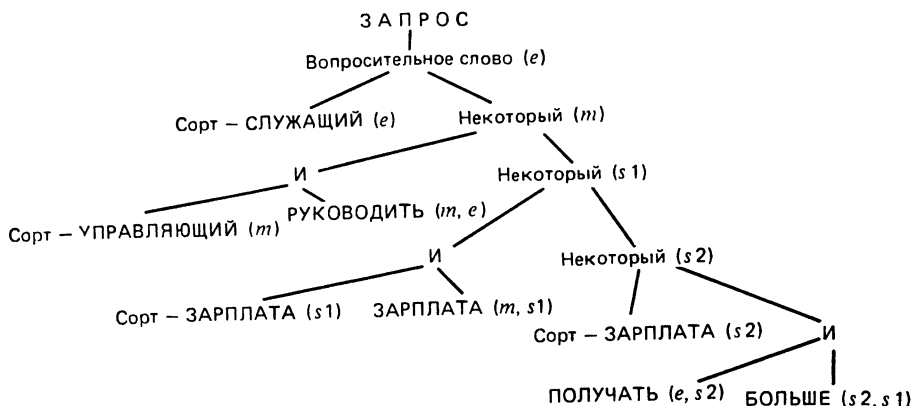


Рис. 2.5. Пример логической формы, полученной в результате анализа запроса

избыточных предикатов, соответствующих одинаковым отношениям (такие предикаты появляются в результате замены виртуальных отношений на отношения, присутствующие в БД); 4) перевод упрощенной в результате предыдущих действий логической формы на язык конкретной БД.

При выполнении этих действий транслятор запросов использует схему БД (см. рис. 2.4), в которой содержатся определения предикатов, используемых в логических формах, в терминах отношений, имеющихся в БД.

Процесс настройки системы ТЕАМ поддерживается блоком приобретения знаний (БПЗ), который обеспечивает получение от прикладного специалиста информации о структуре БД, о словах, обозначающих объекты БД и отношения между ними. Эта информация включается в словарь, концептуальную схему и схему БД. В отличие от общения с пользователем взаимодействие с прикладным специалистом ведется в рамках диалога, инициатива в котором принадлежит системе. На экран терминала выдается одно или несколько меню и вопрос-ответная зона, предназначенная для вывода системой вопросов и ввода прикладным специалистом ответов на эти вопросы.

Основными типами меню являются следующие: 1) меню операций, которые может выполнять БПЗ; 2) меню файлов, содержащее все отношения, о которых известно системе; 3) меню полей, содержащее поля (атрибуты) каждого отношения; 4) меню слов, содержащее все слова из словаря системы. К наиболее важным операциям, выполняемым БПЗ, относятся редактирование типов и определение отношений. Первая операция позволяет вводить новые понятия в иерархию типов — основную структуру данных концептуальной схемы, а вторая — определять связи между виртуальными отношениями и реальными файлами БД.

Вопросы, выводимые системой в вопрос-ответную зону, зависят от выбора прикладным специалистом того или иного пункта в том или ином меню и от ответов на предыдущие вопросы. Так, для ввода новых слов достаточно выбрать пункт new-word в меню слов. Дальнейшее взаимодействие с прикладным специалистом при вводе новых слов показано на рис. 2.6 и 2.7. Полужирным шрифтом выделены ответы, которые выбираются специалистом из предлагаемых системой альтернатив. Как следует из этих примеров, ввод новых слов не требует от прикладного специалиста использования лингвистических терминов. Аналогичный характер имеет взаимодействие между системой и прикладным специалистом и во всех остальных случаях (определение или модификация понятий, отношений, полей и т. п.).

Система IRUS. Структура данной системы (рис. 2.8) во многом подобна структуре системы ТЕАМ. Для выполнения синтаксического и семантического

Введите слово — DEPARTMENT
 Синтаксическая категория — ADJECTIVE NOUN VERB
 Синоним — DEPT
 Множественное число — DEPARTMENTS

Рис. 2.6. Включение в словарь нового существительного

анализа входных запросов на английском языке используется анализатор RUS [Bobrow, 1980], в состав которого входят: программа синтаксико-семантического анализа, являющаяся развитием аналогичной программы в системе SOPHIE; проблемно-независимая ATN-грамматика английского языка; словарь, содержащий как базовые (проблемно-независимые) лексические единицы, так и лексические единицы, вводимые в него при настройке системы.

Семантическая интерпретация выполняется параллельно с синтаксико-семантическим анализом. Это позволяет использовать промежуточные результаты интерпретации для управления синтаксико-семантическим анализом. В результате этих параллельных процессов входные ЕЯ-высказывания преобразуются в формальные выражения на языке представления знаний MRL.

Следующим шагом является расширение полученного MRL-представления в результате разрешения анафорических ссылок и обработки эллиптических конструкций. Для этого используется накапливаемая системой информация о дискурсе — MRL-представления предыдущих запросов.

Расширенное MRL-представление поступает в блок доступа к БД, состоящий из транслятора БД и генератора команд языка манипулирования данными (ЯМД). Транслятор БД осуществляет преобразование MRL-представления на основе информации о логической структуре (схеме) БД. Задачей генератора команд ЯМД является получение одной или последовательности команд ЯМД, позволяющих получить информацию, соответствующую входному запросу.

Процесс настройки системы IRUS, реализуемый системой IRACQ [Moser, 1984], ориентирован не на прикладного специалиста, как в системе TEAM, а непосредственно на конечного пользователя. Предоставляемые пользователю возможности сводятся к вводу новых слов в словарь системы и вводу или коррекции правил семантической интерпретации. Таким образом, настройка системы пользователем ограничена случаями, когда требуется расширение знаний системы (точнее, той ее части, которая выполняет анализ и интерпретацию входных высказываний), необходимых для выражения информационных потребностей пользователей в виде последовательности команд к БД с существующей логической структурой.

В заключение отметим, что, несмотря на очевидные преимущества, предоставляемые пользователю ЕЯ-системами данного класса (по сравнению с формализованными языками СУБД), они в настоящее время еще не получили широкого распространения. На наш взгляд, это объясняется по меньшей мере следующими причинами: сравнительно малое количество коммерческих систем и их

Введите слово — EARN
 Синтаксическая категория — ADJECTIVE NOUN VERB
 Форма 3-го лица, ед. числа, наст. времени (he, she, it) — EARNS
 Прошедшее время — EARNED
 Причастие в прошедшем времени — EARNED
 Пример — AN EMPLOYEE EARNS A SALARY 'A SALARY EARNS' ⇔
 'Something EARNS a SALARY' YES NO
 'AN EMPLOYEE EARNS' ⇔
 'AN EMPLOYEE EARNS something' YES NO
 'A SALARY is EARNED' ⇔
 'Something EARNS a SALARY' YES NO

Рис. 2.7. Включение в словарь глагола

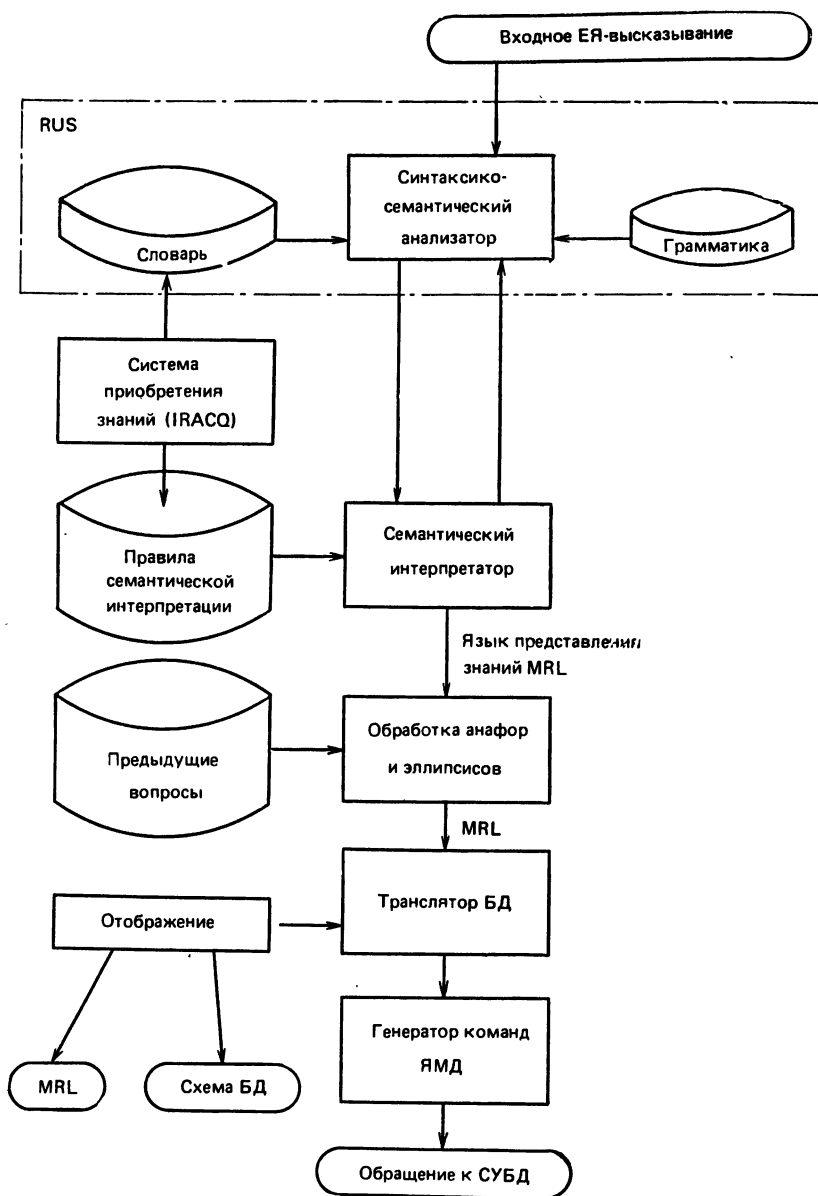


Рис. 2.8. Структурная схема системы IRUS

ориентация на общение преимущественно с реляционными БД; существующие ЕЯ-системы не позволяют задавать запросы, требующие сложной вычислительной и логической обработки, а также не позволяют управлять форматированием отчетов; значительная трудоемкость сопровождения и настройки ЕЯ-систем при их использовании для общения с большими изменяющимися базами данных. Поэтому в настоящее время ЕЯ-системы общения с БД могут применяться лишь в качестве средства оперативного получения информации из каких-то ограниченных областей БД.

2.4. Диалоговые системы решения задач

Основное отличие ЕЯ-систем данного класса от ЕЯ-систем общения с БД состоит в той роли, которую играет система в процессе решения задач пользователя. Системы общения с БД лишь облегчают получение из БД информации, которая требуется пользователю для решения отдельных подзадач стоящей перед ним задачи. Они, как правило, не имеют знаний о задаче, для решения которой пользователю нужна эта информация. Поэтому системы общения с БД играют в процессе решения задач пользователя пассивную роль (за исключением случаев, когда возникают подзадачи, известные системе, например ошибки, незнакомые слова и т. п.), а пользователь — активную.

При общении с диалоговыми системами решения задач пользователь и система меняются ролями. Цель системы состоит в получении решения задачи на основе использования как собственных знаний и механизмов вывода, так и данных, получаемых из ответов пользователя и из прикладных программ, которые могут вызываться диалоговой системой для непосредственного решения каких-то подзадач.

На рис. 2.9 приведен фрагмент диалога [Pazzani et al., 1983], демонстрирующий процесс решения типичной (для ЕЯ-систем рассматриваемого класса) задачи. Данная задача относится к простым задачам информационного обслуживания (см. п. 1.3.2). От системы требуется выполнить или упорядочить для последующего выполнения (т. е. спланировать) действия, позволяющие получить решение некоторой типовой, стереотипной задачи. Каждый класс подобных стереотипных задач характеризуется тем, что входящие в него задачи имеют одинаковую и хорошо определенную структуру и отличаются друг от друга лишь значениями органического числа параметров. Поэтому для инициирования процесса решения задачи пользователю (клиенту) достаточно сообщить системе преследуемую им цель (т. е. идентифицировать тип задачи) и задать ограничения на значения каких-то параметров решаемой задачи (см., например, высказывание П1 на рис. 2.9). Если какие-то параметры пропущены или заданы (с точки зрения системы) неправильно, то система перехватывает инициативу, иницируя диалоги по уточнению параметров (C1, C7). В ходе этих диалогов пользователь также может перехватывать инициативу, задавая системе вопросы (например, П2, П4, П5, П7) для того, чтобы использовать полученные сведения при формировании ответов (П3, П6, П8) на предыдущие вопросы системы.

Предопределенность решаемой задачи и наличие детальных сведений о ее структуре приводят к тому, что основные функции ЕЯ-системы могут успешно выполняться в более сложной (чем в случае ЕЯ-систем общения с БД) постановке. Так, вместо жесткой структуры диалога в диалоговых системах решения задач может использоваться альтернативная или гибкая структура с произвольным переходом инициативы. Понимание входных высказываний осуществляется с учетом текущего состояния диалога и имеющихся у системы целей. Благодаря этому упрощается понимание высказываний, содержащих неправильности, и в то же время повышается непроедурность общения (так как система может рассматривать высказывания пользователя как определения условий текущих подзадач). Высказывания системы строятся в виде фраз естественного языка. Их генерация осуществляется, как правило, в соответствии с коммуникативными намерениями, которые определяются компонентом ведения диалога. В связи

П1: я хочу направить 4 самолета из 707 истребительной эскадрильи
 для удара по Бостону между 11.40 и полуднем
 С1: какие самолеты вы хотите послать?
 П2: какие самолеты имеются в эскадрилье?
 С2: в 707 истребительной эскадрилье F-4S
 С3: вы хотели бы использовать F-4S?
 П3: нет, лучше F-16S из 606 истребительной эскадрильи
 С4: в 606 истребительной эскадрилье нет F-16S
 П4: где имеются F-16S?
 С5: F-16S имеются на аэродромах Кеннеди и Лагардия
 П5: в каких эскадрильях на Лагардии они имеются?
 С6: F-16S имеются в 808 и 909 истребительных эскадрильях
 П6: возьмите их из 808 истребительной эскадрильи на Лагардии
 С7: какая система вооружения должна быть на F-16S?
 П7: какие возможны?
 С8: на F-16S могут быть системы D1, D2 и D4
 П8: ударьте по цели системой D4 и спланируйте остальное
 С9: время достижения цели : 11.50
 время вылета : 11.20
 позывной : PEACE-7

Рис. 2.9. Пример диалогового решения задачи

с этим содержание высказываний системы может в значительной степени варьироваться. Это могут быть результаты решения задач, вопросы, касающиеся каких-то параметров задач, объяснения действий системы и имеющихся у нее представлений о проблемной области и т. п.

Основной областью практического использования диалоговых систем решения задач является, по нашему мнению, обеспечение ЕЯ-доступа к различным прикладным системам, предназначенным для решения задач реальных объемов и сложности. При этом диалоговая система выступает в качестве интерфейса между прикладной системой и конечным пользователем, не знающим входного языка прикладной системы и имеющим лишь самое общее представление об алгоритме решения задачи. В этом случае процесс решения задачи распадается на следующие этапы (мы сейчас не рассматриваем порядок их выполнения): 1) информирование пользователя о возможностях прикладной системы; 2) получение от пользователя исходных данных (описаний задач, подзадач и их параметров), их уточнение и формирование заданий на входном языке прикладной системы; 3) собственно решение подзадач прикладной системой; 4) предоставление пользователю результатов решения задачи.

Большинство из существующих в настоящее время ЕЯ-систем данного типа предназначены для общения пользователя (клиента) с экспертными системами (ЭС) в процессе кооперативного решения задачи. Вместе с тем разрабатываются и ЕЯ-системы, не прибегающие в процессе решения задач к помощи пользователей (эти системы могут найти применение в простых проблемных областях). Ниже рассматриваются четыре ЕЯ-системы диалогового решения задач. Системы SNUKA [Pazzani, 1983], XCALIBUR [Carbonell et al., 1983] и ADVISOR [McKeown et al., 1985] являются типичными представителями ЕЯ-систем общения с ЭС. Последняя из рассматриваемых ниже систем UC [Wilensky, 1982] служит примером системы, не использующей знания пользователя в процессе решения задачи.

Система SNUKA. Данная система обеспечивает общение на английском языке с экспериментальной ЭС KNOBS [Engelman et al., 1979], решающей задачи планирования военных операций (см. рис. 2.9). В функции ЭС входит: проверка совместимости отдельных компонентов плана, перечисление и ранжирование альтернативных компонентов и автоматическая генерация полного плана. KNOBS воплощает модель стереотипного планирования, согласно которой пред-

полагается, что заранее известен некоторый обобщенный план, и задача состоит в получении его конкретной реализации путем присвоения переменным плана согласованных друг с другом значений. (Переменные плана представлены в виде слотов FRL-фреймов.)

Система SNUKA позволяет вводить в ЭС KNOBS компоненты плана на английском языке, а также получать ответы на вопросы пользователя о предметной области, отображаемой в БД системы KNOBS. В процессе анализа и обработки входных высказываний SNUKA использует следующие источники знаний: определения слов английского языка; сценарии, описывающие взаимосвязи между отдельными этапами решения задачи планирования; правила доступа к БД системы KNOBS для получения ответов на вопросы пользователя о предметной области; правила вывода значений пропущенных элементов входных высказываний; правила ведения диалога; правила управления темой диалога.

Анализ входных высказываний в системе SNUKA выполняет анализатор APE-11 [Pazzani et al., 1983], основанный на модели концептуальной зависимости. В процессе анализа используются ожидания, которые могут инициироваться определениями слов, составляющих высказывания, сценариями, а также вопросами, заданными системой (последний вид ожиданий имеет место, если высказывание поступает в ответ на вопрос системы).

Выработка ответов осуществляется на основе правил ведения диалога. Простые правила, например, утверждают, что на вопрос должен быть дан ответ, более сложные правила управляют темой диалога и инициируют или ликвидируют ожидания при изменении темы. Если входное высказывание является вопросом о предметной области (например, «где имеются F-16s?»), то ответ формируется по правилам, содержащим три элемента: условие, представляющее собой образец (выраженный в терминах модели концептуальной зависимости), который связывает переменные с именными группами вопроса; действие, которое используют референты этих именных групп для генерации запроса к БД системы KNOBS; ответ, являющийся шаблоном предложения на английском языке, в который подставляется информация, полученная из БД.

В системе SNUKA сценарии являются описаниями стереотипных планов достижения целей (например, проведения военных операций). Роли сценария (т. е. объекты и участники, составляющие его контекст) соответствуют переменным планов в системе KNOBS. Важное отличие сценариев от фреймов системы GUS [Bobrow et al., 1977a] состоит в том, что сценарии являются совокупностями отдельных сцен, между которыми установлены причинно-следственные и временные отношения. Каждая сцена описывает взаимосвязи между состояниями и действиями, которые могут происходить в соответствующем контексте. Эта информация используется для преобразования запросов пользователя (т. е. приведения их в соответствие с существующими в предметной области отношениями между объектами и участниками), выделения значений переменных из входных высказываний, определения вопросов, задаваемых системой пользователю, и порядка, в котором они должны быть заданы, а также для выбора смысла слов и обработки анафорических местоименных отсылок при анализе входных высказываний.

Система XCALIBUR. Данная система позволяет общаться на английском языке с ЭС XSEL [McDermott, 1982a]. Система XSEL вместе с XCALIBUR выполняет функции консультанта, помогающего пользователю (клиенту) выбрать нужные ему компоненты вычислительной VAX и формирующего с помощью системы R1 [McDermott, 1980] заказ на конфигурацию технических средств (рис. 2.10). К основным типам входных высказываний, поступающих в XCALIBUR, относятся: 1) вопросы о характеристиках отдельных технических средств; 2) требования (команды) на включение того или иного технического средства в заказ; 3) ответы на уточняющие вопросы системы.

Основными компонентами XCALIBUR являются: анализатор английского языка DYPAR-II; блок управления; генератор фраз английского языка. DYPAR-II относится к классу анализаторов, использующих разнообразные методы (см. § 1.3). Процесс анализа входных высказываний объединяет: рекурсивное озна-

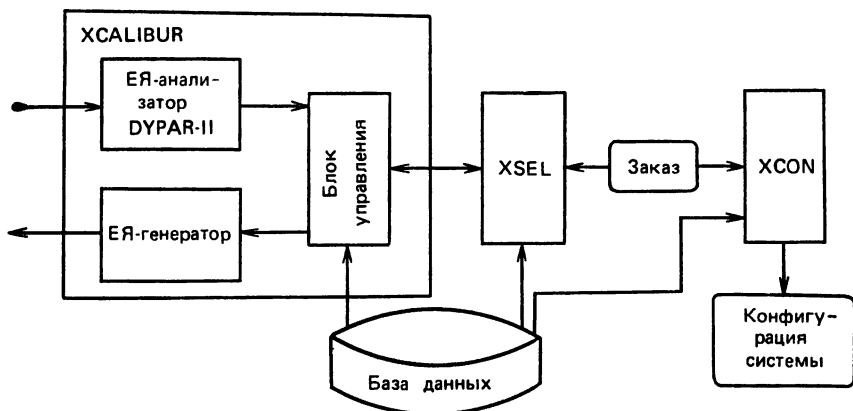


Рис. 2.10. Основные компоненты системы XCALIBUR

чивание фреймов, использование семантической грамматики, сопоставление по образцу и глобальные трансформационные стратегии. Это позволяет повысить гибкость, мощность и выразительность языка общения до уровня, недостижимого в случае использования анализаторов, основанных на каком-нибудь одном методе.

Фрейм системы XCALIBUR состоит из заголовка-образца и множества именованных слотов (падежей), содержащих синтаксические и семантические ограничения (ожидания), которым должны удовлетворять значения-заполнители этих слотов. Падежи и фреймы соответствуют семантическим единицам проблемной области. Несмотря на то, что различные типы фреймов, например «глагол-действие» или «сложное существительное», играют различные синтаксические роли, все они распознаются с помощью одного и того же механизма.

Эффективность и мощность анализатора достигается в результате комбинирования восходящего процесса распознавания семантических образов (соответствующих, например, заголовкам падежных фреймов) с нисходящим процессом заполнения (означивания) падежей, управляемым ожиданиями. При этом могут рассматриваться не только ожидания, соответствующие падежам означиваемого в текущий момент фрейма, но и ожидания, учитывающие характер предыдущих высказываний. Использование ожиданий для управления процессом заполнения падежей позволяет успешно преодолевать ошибки во входных высказываниях, а также эффективно обрабатывать разговорные конструкции, такие, как, например, эллипсис.

Блок управления является посредником между ЕЯ-анализатором, экспертной системой и ЕЯ-генератором. В тот момент, когда система готова принять задание, блок управления получает и передает анализатору извещение, которое идентифицирует фрейм (или класс фреймов), задающий ожидаемую структуру задания. Пусть, например, ожидается команда или запрос и пользователь вводит: «Сколько стоят 2 наибольших двухпортовых диска?». В результате анализа этого высказывания (с учетом переданного анализатору ожидания) будет получено следующее представление:

ЗАПРОС (ЦЕЛЬ (НАЙТИ (диск
(порты (ЗНАЧЕНИЕ (2)))
(тип диска (ЗНАЧЕНИЕ (НЕСЪЕМНЫЙ)))
(ОПЕРАЦИЯ (СОРТИРОВАТЬ
(ТИП (по убыванию))
(АТРИБУТ (размер)))

(КОЛИЧЕСТВО (2)))
(ВЫДАТЬ (цена)))
(ИСТОЧНИК ИНФОРМАЦИИ (неопределен))))))

Это представление выражает информацию, т. е. выделенные в результате анализа запроса значения слотов ожидаемого фрейма в терминах атрибутов, используемых в программах блока управления и адапторе системы XSEL.

Блок управления присваивает значения атрибутам по умолчанию, добавляет атрибуты в соответствии с требованиями программ ЭС, выполняющих команды пользователя, а также отображает неоднозначные имена атрибутов (например, «размер» или «скорость») в имена, соответствующие понятийной структуре ЭС. Кроме того, блок управления включает в представления высказываний пользователя дополнительные функции. Так, в большинстве случаев, когда требуется диск емкостью 300 Мбайт, это означает, что диск должен иметь емкость не менее 300 Мбайт.

Процесс генерации выходных высказываний включает три этапа: ответ преобразуется в граф концептуальной зависимости; выбирается глагол, и граф концептуальной зависимости отображается в соответствующий падежный фрейм; падежный фрейм трансформируется в ЕЯ-высказывание. Первый этап является проблемно-зависимым. Второй этап выполняется программой, которая аналогична применяемой в синтезаторе BABEL [Шенк, 1980] системы MAPGE. При выполнении третьего этапа помимо традиционных средств построения поверхностной структуры фраз английского языка используются средства формирования местоименных ссылок на ранее встречавшиеся именные группы.

Система ADVISOR. Данная система создавалась с целью развития объяснительных возможностей ЕЯ-систем, обеспечивающих взаимодействие пользователя с ЭС. ADVISOR выполняет функции консультанта, способного отвечать на английском языке на вопросы студентов о различных дисциплинах (курсах), изучаемых в университете, и давать советы (объяснения) относительно возможности или необходимости изучения той или иной дисциплины.

Основными функциональными компонентами системы ADVISOR являются: АТН-анализатор; база знаний, реализованная средствами KL-ONE [Brachman, 1979], и функции доступа к ней; блок вывода целей; генератор ответов и объяснений [Derg et al., 1984] (рис. 2.11). Сопряженная с ADVISOR ЭС вызывает ся при поступлении вопросов, требующих от системы совета. Эти вопросы подразделяются на два класса: 1) «can»-вопросы, например «Can I take a discrete math this semester?» (Могу ли я изучить дискретную математику в этом семестре?). 2) «should»-вопросы, например «Should I take data structures?» (Должен ли я изучать структуры данных?) В результате обработки этих вопросов, переданных ЭС в терминах внутреннего представления, ЭС возвращает генератору не только ответ («да» или «нет»), но и информацию (последовательность использованных продукций), необходимую для генерации соответствующего объяснения.

Отличительная особенность данной системы — наличие специального механизма установления (распознавания) целей (намерений) пользователя, относящихся к обсуждаемой проблемной области. Установив цели спрашивающего, система выбирает и использует для формирования ответов (объяснений) ту информацию из базы знаний, которая соответствует этим целям.

Например, один и тот же ответ на вопрос: «Должен ли я изучать вместе дискретную математику и структуры данных в этом семестре?» — может сопровождаться в зависимости от целей спрашивающего различными объяснениями: 1) «Да, структуры данных требуются для изучения всех дальнейших дисциплин, а дискретная математика является сопутствующей дисциплиной»; 2) «Да, их обычно изучают вместе на первом семестре второго курса»; 3) «Да, если изучить структуры данных в этом семестре, то в следующем можно изучать введение в ИИ, а дискретную математику нужно изучать одновременно со структурами данных».

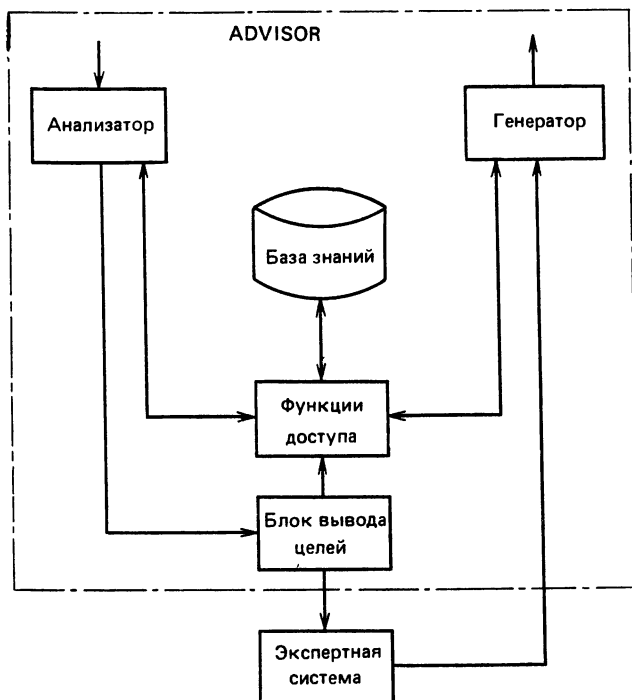


Рис. 2.11. Структурная схема системы ADVISOR

База знаний ADVISOR может быть представлена в виде семантической сети, разбитой на пересекающиеся подпространства, которые определяют отношения между сущностями и событиями проблемной области, отражающие тот или иной взгляд (или точку зрения) пользователя. Таким образом, один и тот же факт или событие может рассматриваться системой как бы с различных точек зрения, соответствующих возможным целям пользователя. Это позволяет производить интерпретацию и дальнейшую обработку запроса по-разному, т. е. используя информацию из различных подпространств семантической сети в зависимости от цели спрашивающего, выражаемой его запросом (или последовательностью запросов).

Распознавание целей осуществляется блоком вывода целей. В системе ADVISOR различаются два вида целей: текущие и релевантные. Текущие цели определяются для каждого высказывания пользователя. Их распознавание осуществляется на основе применения методологии, изложенной в [Аллен и др., 1986].

Под релевантными целями понимаются цели, выражаемые совокупностью высказываний, составляющих некоторый фрагмент диалога. Релевантные цели выводятся из текущих следующим образом. После установления текущей цели (по начальному высказыванию пользователя) система с помощью правил типа «состав-действие» (эти правила постулируют, что если пользователь хочет выполнить действие *A*, которое является частью действия *B*, то он может хотеть выполнить действие *B*) пытается построить множество целей более высокого уровня. Получаемые иерархии целей называются планами. Например, при поступлении запроса «Предлагается ли курс по ЕЯ в следующем семестре?» будет

- П1: как удалить файл?
 С1: команда 'RM<имя файла>' удалит файл с именем
 <имя файла> из вашей текущей директории
 П2: переименовать файл?
 С2: команда 'MV<имя файла><новое имя файла>' переименует
 файл с именем <имя файла> в файл с именем <новое имя файла>
 П3: как создать файл в UNIX?
 С3: наилучший способ создать файл – это воспользоваться
 редактором, т. е. командой 'EX<имя файла>'
 П4: директорию?
 С4: команда 'MKDIR<имя директории>' создаст директорию
 с именем <имя директории> в текущей директории
 П5: как узнать, какие пользователи работают в системе?
 С5: команда 'WHO' позволит узнать, кто связан с системой

Рис. 2.12. Пример работы системы UC

установлена текущая цель «Изучение дисциплины ЕЯ», а целями более высокого уровня, выведенными из данной, могут быть «Специализация в области ИИ» и «Изучение факультативных дисциплин».

Такие же действия выполняются при поступлении следующих запросов, например «Кто читает лекции по ИИ?». В данном случае текущая цель «Изучение ИИ», а выведенные цели более высоких уровней: «Специализация в области ИИ» и «Изучение обязательных дисциплин». Релевантная цель получается в результате пересечения планов, выведенных из текущих целей отдельных высказываний. В рассматриваемом случае в качестве релевантной цели будет получено: «Специализация в области ИИ».

Процесс вывода релевантной цели продолжается до поступления «should» или «can» вопроса. Соответствующая релевантной цели информация, содержащаяся в базе знаний (она относится к одному из подпространств семантической сети), передается в рабочую память ЭС, сопряженной с ADVISOR. Выходом ЭС является ответ на вопрос, т. е. «да» или «нет», а также исходные данные для генерации поверхностной структуры объяснения.

Система UC. В отличие от предыдущих ЕЯ-систем диалогового решения задач данная система получает ответы на вопросы пользователя, используя только собственные знания. Система UC выполняет функции консультанта по операционной системе UNIX. Общение с пользователем ведется на английском языке. Цель общения состоит в том, чтобы объяснить пользователю, какие команды операционной системы должны быть применены в каждой конкретной ситуации. Типичный пример диалога с этой системой приведен на рис. 2.12. Из этого примера следует, что диалог с системой UC имеет много общего (с точки зрения формы диалога, используемых языковых конструкций и т. п.) с диалогами, поддерживаемыми системами общения с БД. Однако в отличие от систем общения с БД UC (и аналогичные ей системы) не осуществляет перевода запросов пользователя в соответствующие команды UNIX, готовые к исполнению, а выполняет функции консультанта, подсказывающего синтаксис и семантику команд, применимых в ситуации, описываемой в запросе пользователя.

Система UC может служить примером ЕЯ-системы, собранной из ранее разработанных фрагментарных систем, которые используются помимо UC и в других ЕЯ-системах. Основными компонентами UC являются: анализатор PHRAN [Wilensky, 1981a], генератор PHRED [Jacobs, 1983], блок вывода PANDORA [Wilensky, 1981b]; база знаний, реализованная на основе системы представления PEARL [Deering et al., 1981] (рис. 2.13).

Анализатор PHRAN, так же как и генератор PHRED, первоначально использовались в системе обработки связанных текстов FAUSTUS (см. ниже). Для перевода входных высказываний во внутреннее представление применяется механизм сопоставления по образцу. При включении PHRAN в диалоговую систему UC

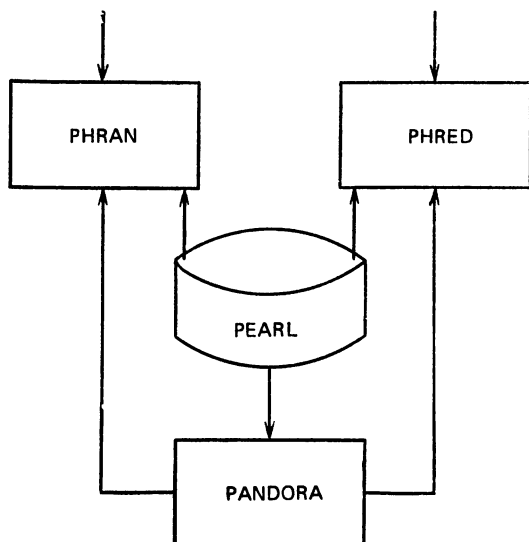


Рис. 2.13. Структура системы UC

потребовалось его расширение для того, чтобы обеспечить понимание ряда разговорных конструкций, например различных видов эллипсисов.

Внутреннее представление входных высказываний передается в блок вывода PANDORA. Задачей этого блока является получение внутреннего представления выходного высказывания. В простейшем случае PANDORA использует внутреннее представление вопроса пользователя в качестве формального запроса к базе знаний для непосредственного поиска фактов, которые могли бы служить ответами на этот вопрос. Однако большинство реальных вопросов требует более сложной обработки. Например, буквальный ответ на косвенный вопрос «Мне хотелось бы получить пространство на диске» мог бы иметь вид: «Введите *mt*» (т. е. система

рекомендует использовать команду, приводящую к удалению файла). Очевидно, что если пользователь хотел получить дополнительное пространство (т. е. не удаляя своих файлов), то данный ответ является неудовлетворительным.

При обработке подобных вопросов PANDORA пытается вывести неявные цели пользователя и в случае их насовпадения с целями, явно указанными в вопросе, сформировать план по устранению противоречий между ними. Для вывода неявных целей PANDORA сначала моделирует состояние, которое будет получено в результате выполнения имеющегося в базе знаний плана достижения явной цели (в приведенном выше примере этот план предусматривает применение команды удаления файла). Далее анализируются различия между текущим и смоделированным состояниями. Эти различия разбиваются на две группы. К первой относятся различия, сигнализирующие о достижении явной цели. Вторую группу составляют различия, обусловленные побочными эффектами. Например, при выводе неявных целей для приведенного выше вопроса к побочным эффектам будет отнесено удаление какого-то из файлов пользователя.

Для каждого побочного эффекта генерируется цель по его устранению, которая рассматривается как возможная неявная цель, вступающая в конфликт с явной целью. Так, для приведенного выше вопроса возможная неявная цель состоит в сохранении существующих на диске файлов. Однако наличие конфликта в данном случае является лишь возможным, поскольку у пользователя могут быть ненужные файлы. В подобных ситуациях PANDORA проверяет существование конфликта на самом деле. Для этого генерируется вопрос: «Есть ли у Вас ненужные файлы?» Если ответ пользователя отрицательный, то конфликт действительно существует и должна быть использована стратегия разрешения конфликтов.

Общая стратегия, которая применима ко всем конфликтам, вызванным нехваткой какого-либо ресурса, состоит в попытке получения дополнительного ресурса. В рассматриваемом примере недостающий ресурс — это пространство на диске. Поэтому PANDORA создает новую цель, состоящую в получении дополнительного (по отношению к уже имеющемуся у пользователя) пространства. План достижения этой цели предусматривает обращение к администратору си-

стемы, поэтому УС выдает в качестве ответа соответствующую рекомендацию.

Из рассмотрения работы блока вывода системы УС следует, что ее диалоговые возможности (в частности, возможности перехвата инициативы) зависят от заложенных в базу знаний планов достижения различных целей. Например, если бы план достижения явной цели в рассмотренном примере предусматривал обращение к администратору системы, а не использование команды удаления файлов, то можно было бы обойтись без вывода неявных целей и последующего разрешения конфликта. Однако такое решение исключило бы возможность информирования пользователя о том, что он мог бы сначала удалить ненужные ему файлы. Таким образом, применение более «мелких» единиц для представления планов достижения целей ведет к повышению гибкости процесса общения.

Завершая рассмотрение ЕЯ-систем диалогового решения задач, отметим, что существующие ЕЯ-системы данного класса пока не отвечают требованиям, диктуемым условиями промышленной эксплуатации. В частности, мы имеем в виду требование простоты настройки ЕЯ-системы на класс решаемых задач и на прикладную систему (например, ЭС), используемую в ходе решения задачи. В то же время следует обратить внимание на тенденцию применения в качестве подсистем ЕЯ-системы хорошо зарекомендовавших себя и допускающих настройку фрагментарных систем. Эта тенденция особенно заметна в ЕЯ-системах, решающих задачи с помощью собственных механизмов вывода. Так, в рассмотренной выше системе УС используются анализатор PHRAN, который может быть настроен на английский или испанский язык, и генератор PHRED, настраиваемый на английский, испанский и китайский языки. Учитывая широкую распространенность операционной системы UNIX, следует, по-видимому, ожидать в ближайшее время появления коммерческой версии системы УС.

2.5. Системы обработки связанных текстов

Системы данного класса моделируют процесс понимания законченных описаний каких-то фрагментов действительности (историй, рассказов, эпизодов и т. п.), выраженных в виде текста на естественном языке, т. е. последовательности связанных друг с другом предложений. Понимание текста трактуется как извлечение из него всей существенной с точки зрения системы информации и присоединение ее к собственной базе знаний. После этого система может отвечать на вопросы относительно фактов, событий, явлений и прочих сущностей, которые явно или косвенно описаны во введенных текстах. Очевидно, что в практическом плане модели и методы, развиваемые в системах обработки связанных текстов, могут быть полезны при создании интеллектуальных систем автоматического индексирования и реферирования.

Типичные приемы и методы, применяемые в системах обработки связанных текстов, продемонстрируем на примерах четырех систем RESEARCHER [Lebowitz, 1985], TAILOR [Макьюин, 1986], FAUSTUS [Norvig, 1983] и TACC [Пранц, 1983]. Прежде чем перейти к их рассмотрению, отметим, что если ЕЯ-системы общения с БД уже вышли на уровень коммерческого или промышленного продукта, а выход на этот уровень ЕЯ-систем диалогового решения задач ожидается в ближайшем будущем, то системы обработки связанных текстов находятся на стадии разработки экспериментальных образцов, которые используются для исследования и оценки методов решения этой крайне сложной и многогранной задачи.

Системы RESEARCHER и TAILOR. Данные системы образуют единый комплекс (рис. 2.14), позволяющий пользователю получать сведения из рефератов патентов, описывающих сложные физические объекты. Система RESEARCHER получает рефераты патентов (на английском языке), строит на их основе базу знаний и делает обобщения из различных патентов, которые могут служить для изучения содержащихся в рефератах сведений, относящихся к различным объектам. Вопрос-ответные функции выполняет система TAILOR, которая будет рассмотрена ниже.

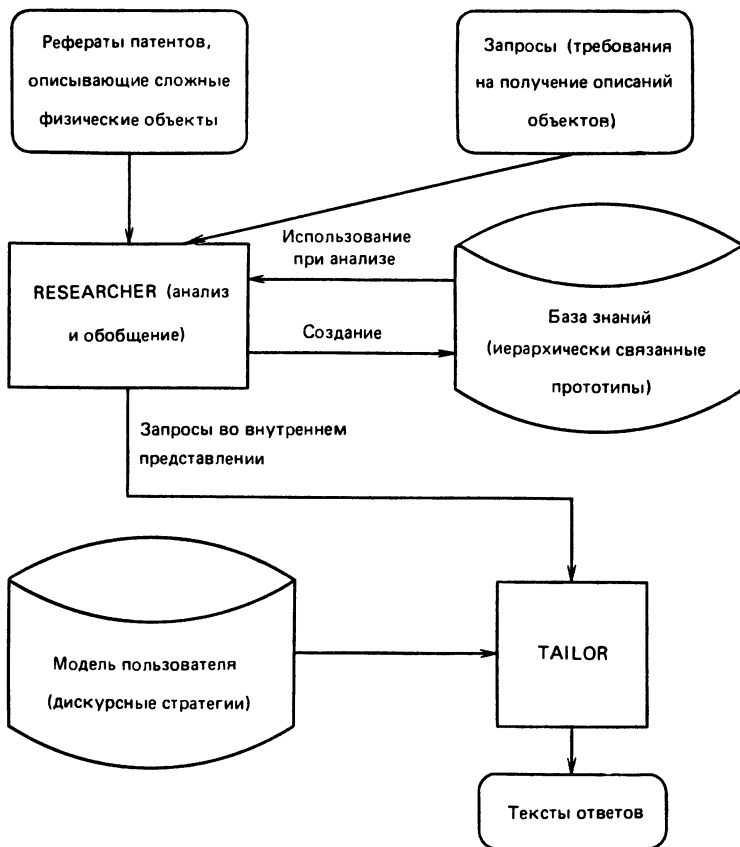


Рис. 2.14. Схема взаимодействия систем RESEARCHER и TAILOR

Понимание входных текстов в системе RESEARCHER основано на методах концептуального анализа, ранее успешно применявшихся в таких системах, как MARGE, FRUMP, IPP и т. д. В отличие от этих систем в данном случае входными текстами являются описания сложных физических объектов, а не событий (например, IPP «понимает» описания происшествий, публикуемые в газетной хронике). Поэтому для RESEARCHER была разработана специальная фреймподобная система представления знаний, ориентированная не на действия, которые могут производиться участниками событий, а на объекты [Wasserman et al., 1983].

Внутреннее представление рефератов патентов в системе RESEARCHER включает следующие виды информации: 1) иерархические описания, отражающие деление объектов на компоненты; 2) физические и функциональные отношения между компонентами; 3) свойства объектов. Внутреннее представление каждого нового реферата объединяется с уже накопленной базой знаний с помощью механизма обобщения иерархических описаний. Он позволяет получать обобщенные объекты (прототипы), которые могут быть иерархически связаны друг с другом, и при поступлении описания нового объекта относить его к тому

или иному прототипу и запоминать лишь параметры, по которым этот объект отличается от прототипа.

Система TAILOR предназначена для генерации текстов в ответ на запросы пользователей с различной квалификацией к базе знаний, сформированной RESEARCHER. Степень детальности описаний объектов в генерируемых текстах определяется исходя из модели пользователя и поэтому может быть различной.

TAILOR состоит из двух основных компонентов: стратегического, определяющего, какая информация из базы знаний должна быть включена в ответ, и тактического, трансформирующего сообщение, сформированное стратегическим компонентом, на английский язык.

Работа стратегического компонента основана на использовании так называемых дискурсных стратегий [Макьюин, 1986], составляющих модель пользователя и определяющих структурные свойства порождаемых текстов. В рамках дискурсных стратегий учитываются цель порождения текста (например, дать определение, описать, сравнить, подтвердить и т. п.) и знания пользователя в обсуждаемой предметной области. При разработке обобщенных моделей текстов, соответствующих дискурсным стратегиям, учитывались приемы, которыми пользуются для составления словарных статей о различных физических объектах в энциклопедиях и справочниках, предназначенных для специалистов и неискушенных читателей [Paris, 1985].

Так, дискурсная стратегия «описать», ориентированная на специалистов, определяет модель порождаемого текста в виде дерева составляющих, которое включает следующие компоненты: отнесение объекта к какому-либо обобщенному классу; представление составляющих описываемого объекта; представление информации, характерной для каждой составляющей объекта; представление дополнительной информации об описываемом объекте. Используя данную стратегию при обработке запроса на описание телефона, TAILOR генерирует следующий текст:

«Телефон — это устройство. Телефон состоит из микрофона, корпуса, шнура и приемника; микрофон имеет двухрезонансную систему и диафрагму. Корпус — это своего рода кожух. Шнур — это провод. У приемника есть мембрана, воздушный зазор и электромагнит...»

Кроме того, дискурсная стратегия, ориентированная на неспециалистов (в частности, детей), предусматривает построение текста в виде описания процесса, приводящего объект в действие. Так, получив тот же вопрос на описание телефона, TAILOR выдает следующий ответ неспециалисту:

«Когда абонент говорит, звуковые волны попадают на мембрану. При этом мембрана колеблется точно так же, как молекулы воздуха...»

Таким образом, стратегический компонент с помощью дискурсных стратегий может определять, «что сказать», т. е. какая информация об объектах должна быть включена в порождаемый текст ответа. Тактический компонент решает, «как сказать», т. е. синтезирует поверхностную структуру текста ответа. Для решения этой задачи используется функциональная грамматика унификации [Кай, 1979] и словарь. Словарь позволяет произвести выбор глагола и его аргументов (которые в дальнейшем будут трансформированы в подлежащее и дополнение предложения) для каждого из предикатов внутреннего представления генерируемого текста. В результате получается промежуточное представление текста, имеющее древовидную структуру, но в котором (в отличие от внутреннего представления) известны все слова для будущего ЕЯ-предложения.

Последний шаг генерации (он называется унификацией) связан с применением правил грамматики, которые представлены в том же формализме, что и текст. В ходе этого процесса строится синтаксическая структура предложения, производится выбор пассивного или активного залога, в результате чего древовидная структура внутреннего представления линейруется и выходное предложение приобретает окончательный вид.

Система FAUSTUS. Особенностью данной системы обработки связных текстов является применение фреймов в качестве универсального средства представления всех классов знаний (событий, объектов, планов и целей участников со-

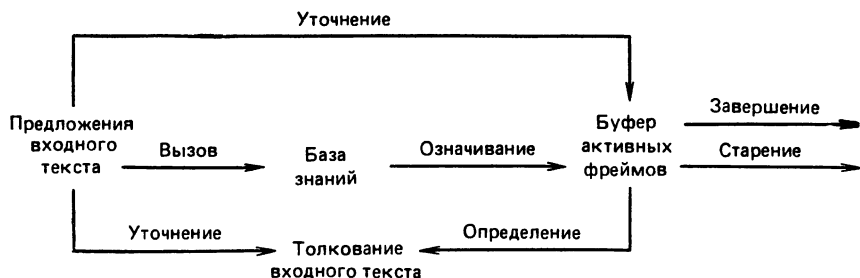


Рис. 2.15. Основные процессы, обеспечивающие понимание текстов в системе FAUSTUS

бытий, особенностей сюжета и т. п.), необходимых для понимания текста. Это позволяет использовать единый механизм интерпретации на всех фазах обработки текстов и за счет этого уменьшить зависимость системы от специфики содержания текстов.

Преобразование входных высказываний (т. е. вводимых в систему текстов и вопросов относительно содержания этих текстов) осуществляется анализатором PHRAN, реализующим метод концептуального анализа, обратное преобразование ответов системы из внутреннего представления — генератором PHRED. Собственное понимание текста как последовательности связанных друг с другом предложений происходит в результате работы механизма интерпретации. Интерпретация включает шесть основных процессов, выполняемых над фреймами: вызов; означивание; определение; уточнение; завершение; старение (рис. 2.15).

Обработка текста начинается с того, что PHRAN получает его внутреннее представление, которое используется для поиска в базе знаний «подходящих» для этого предложения фреймов. База знаний представляет собой организованную в виде фреймов совокупность утверждений о проблемной области и ассоциированных с ними понятий. Если какой-то из элементов внутреннего представления входного предложения сопоставляется (т. е. подходит в качестве значения) с элементом некоторого фрейма, то происходит вызов фрейма. Каждому вызову приписывается вес, и когда суммарный вес вызовов фрейма превосходит определенный порог, этот фрейм означивается, т. е. создается пример фрейма, и его слоты принимают значения из входного предложения.

Множество означенных фреймов сохраняется в буфере активных фреймов (БАФ), определяющем контекст, с учетом которого обрабатываются последующие предложения текста. При этом элементы следующих предложений могут интерпретироваться как значения слотов активных фреймов. Данный процесс называется уточнением. Внутреннее представление текста, называемое толкованием, формируется в результате выбора из БАФ на основании ряда эвристик наиболее «подходящих» фреймов. Процесс выбора называется определением, а фреймы, составляющие толкование, — определяющими фреймами. Поскольку содержание БАФ зависит от последующих предложений текста, FAUSTUS имеет два процесса, предназначенных для удаления нерелевантных фреймов из БАФ. Первый процесс — завершение — противоположен определению: удаление активного фрейма, если установлено, что он по каким-то причинам не подходит. И наконец, если фрейм в течение определенного промежутка времени не получил статус определяющего (т. е. не вошел в толкование), то процесс старения исключает его из БАФ.

Толкование текста, полученное в результате интерпретации, представляет собой совокупность взаимосвязанных примеров фреймов, описывающих каждое действие, о котором говорится (явно или неявно) в тексте, как элемент плана того или иного участника действия по достижению определенной цели. Поэтому в ответы на вопросы пользователя легко может включаться информация, явно-

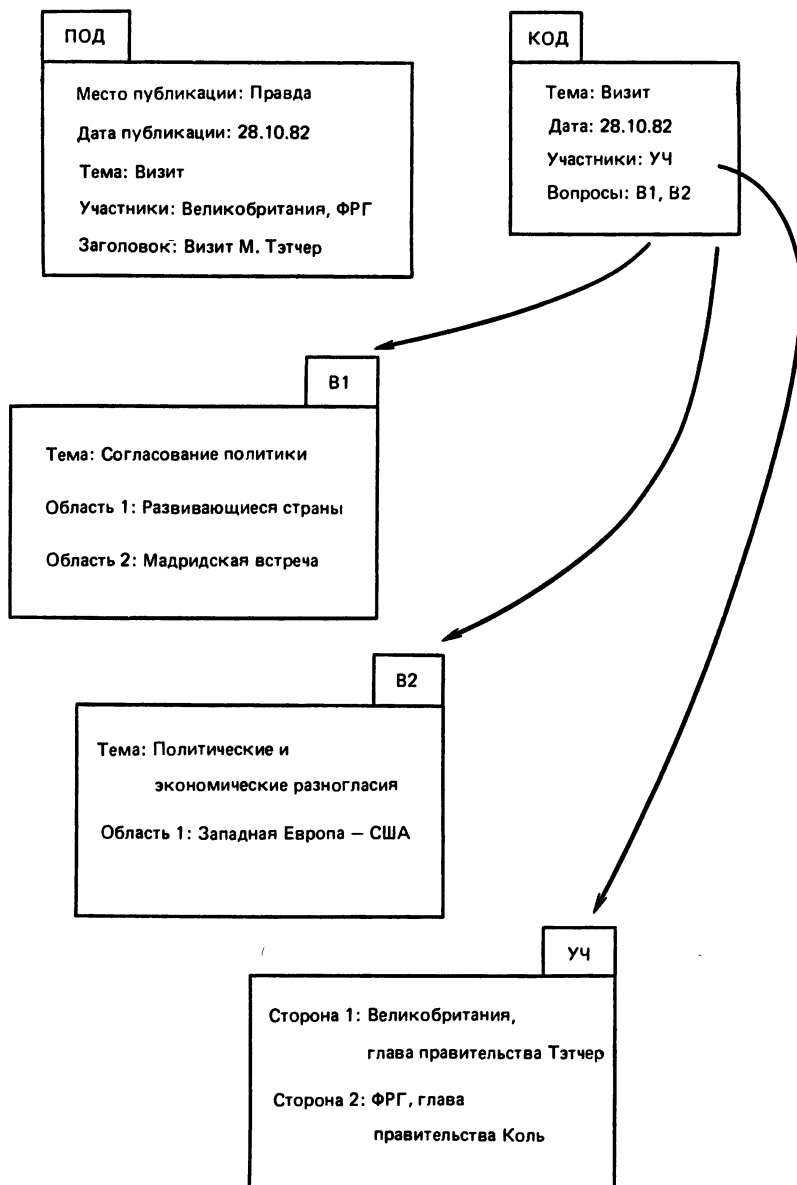


Рис. 2.16. Результат работы системы ТАСС

не содержащаяся во введенных текстах и имеющая характер объяснений поступков участников событий, о которых говорится в исходных текстах.

Система ТАСС. Данная система предназначена для автоматизированной обработки газетных сообщений на русском языке. Основными функциями системы являются: 1) формирование поисковой информации, характеризующей газетный текст в целом, и формирование модели предметной области; 2) поиск газетных сообщений, содержащих интересующие пользователя факты; 3) исследование модели предметной области с целью анализа, уточнения и прогнозирования сложившейся ситуации. Непосредственное отношение к теме настоящего параграфа имеет только первая из указанных функций, поэтому вторая и третья функции не рассматриваются.

В отличие от ранее рассмотренных систем в системе ТАСС обработка связанных текстов выполняется автоматизированно, т. е. с участием человека. Автоматизированная обработка текста опирается главным образом на богатство лексических сведений и поддерживается возможностью диалогового общения в тех ситуациях, когда лингвистические знания ТАСС оказываются недостаточными. В этом случае предметом исследования для системы становится сложившееся у человека представление о содержании текста, а диалог преследует две цели: построение внутреннего представления текста и уточнение модели предметной области.

База данных системы ТАСС, называемая объектной базой (ОБ), состоит из четырех разделов: Понятия, Факты, Имена, Тексты. С каждым разделом связаны свои процессы обработки информации, в частности процессы, которые организуют уточняющие диалоги. Объекты, входящие в раздел Понятия, представляют собой описание (в виде системы фреймов) модели предметной области и служат прообразами объектов с фактической информацией. При этом в зависимости от того, какое имя (собственное или нарицательное) получит объект в процессе конкретизации фреймов-понятий при анализе исходных текстов, он попадает в разделы Имена или Факты соответственно.

Два объекта-факта ПОД и КОД (вместе с объектами, на которые имеются ссылки) задают соответственно поисковый и концептуальный образы исходного текста, используемые при поиске статей и содержащихся в них фактов по запросам пользователя. Следует отметить, что со значением некоторых свойств объектов-фактов могут быть связаны коэффициенты достоверности, выражающие степень уверенности в том, что атрибут данного объекта имеет указанное значение. Это позволяет хранить несколько (в том числе противоречивых) значений одного и того же атрибута. Наконец, тексты газетных сообщений вместе с индексами, задающими самую общую характеристику публикаций, составляют четвертый раздел ОБ — Тексты. Примеры объектов, полученных системой ТАСС в результате обработки сообщения газеты «Правда» о визите М. Тэтчер в ФРГ, приведены на рис. 2.16.

Завершая рассмотрение данного класса ЕЯ-систем, следует подчеркнуть, что задача понимания связанных текстов в общем случае превосходит по сложности задачи, решаемые ЕЯ-системами ранее рассмотренных классов. Наиболее сложными для понимания являются тексты, описывающие взаимоотношения и поступки активных действующих лиц. Для их понимания ЕЯ-система не только должна обладать громадным объемом знаний о мире, но и иметь совершенные механизмы вывода, позволяющие ей определять мотивы этих поступков, предвидеть намерения действующих лиц и т. п. Такой уровень понимания текста, по-видимому, не может быть достигнут в обозримом будущем. Поэтому с прикладной точки зрения представляется оправданным использование человеко-машинного подхода к решению этой задачи (см. система ТАСС).

Сложность задачи понимания текстов уменьшается, если тексты содержат описание фрагментов статического мира, а количество отношений между сущностями этого мира невелико (см. систему RESEARCHER). На наш взгляд, именно такие области будут «освоены» в первую очередь промышленными ЕЯ-системами данного класса.

Отечественные коммерческие и промышленные ЕЯ-системы

3.1. Коммерческая система АИСТ

А. Б. Преображенский

3.1.1. Общее описание. Система предназначена для обеспечения доступа неподготовленным конечным пользователей к БД, поддерживаемым СУБД типа ДИСОД [Система ДИСОД, 1987]. Функциональные возможности АИСТ реализуются в рамках диалоговых приложений — типовых задач, совместно решаемых пользователем и системой в процессе диалога, имеющего гибкую структуру (т. е. АИСТ обладает свойствами как ЕЯ-систем общения с БД, так и ЕЯ-систем диалогового решения задач). По своему назначению типовые задачи подразделяются на основные, решение которых направлено на удовлетворение информационных потребностей пользователя относительно проблемной области (ввод документов в БД, поиск и выдача документальной, фактографической и документально-фактографической информации из БД), и технологические (вспомогательные), предусматривающие расширение и модификацию функций системы (создание новых типовых задач и ввод новой лексики).

Система АИСТ рассчитана на широкий класс практических применений. Она может использоваться при решении управленческих задач и задач информационно-справочного обслуживания в экономике, медицине, бухгалтерском учете, материально-техническом снабжении, библиотечном деле, делопроизводстве, и многих других областях. Во всех приложениях проблемная область отображается в БД ДИСОД в виде совокупности файлов, имеющих в общем случае иерархическую структуру. Каждую запись (или совокупность связанных записей) БД, соответствующую определенной сущности (объекту, факту, событию и т. п.) проблемной области, удобно представить в виде документа, описывающего различные свойства этой сущности. Документы, вводимые в БД, и запросы на поиск документов могут быть представлены в этом случае в виде фиксированных форматов (анкет), состоящих из списка пар имя реквизита — выражение (фактографическое или содержательное). При этом одни реквизиты (фактографические) содержат фактографические сведения о документе (например, в библиотечном деле: автор, издательство, год издания, количество страниц, код по УДК, адрес хранения, количество требований и т. п.), а другие выражают собственно содержание документа (например: название, аннотация, реферат и т. п.).

Система АИСТ обеспечивает перевод документов и запросов во внутреннее представление. При этом фактография выражается в виде слов и словосочетаний русского языка (значений фактографических реквизитов), связанных фиксированными отношениями (или, и, от ... до, меньше, больше, кроме, не равно, раньше, позже и т. д.), а содержательные выражения представляются в виде текстов русского языка без каких бы то ни было ограничений. Это позволяет во всех приложениях этой системы обеспечить широкому классу неподготовленных конечных пользователей доступ на русском языке в анкетной (табличной) форме к информации, хранящейся в БД.

Основными компонентами АИСТ (рис. 3.1) являются компоненты ведения диалога, понимания высказываний и генерации высказываний. Компонент ведения диалога обеспечивает реализацию возможностей, предоставляемых АИСТ, в виде типовых задач, решаемых совместно с пользователем. Процесс решения каждой типовой задачи определяется заранее заготовленным сценарием, который представляет собой частично упорядоченное множество правил с параметрами, значения которых устанавливаются в ходе решения конкретной задачи. При этом в качестве значений параметров могут использоваться внутренние представления высказываний пользователя и информация о проблемной области, содержащаяся в БД, общение с которой поддерживается с помощью системы

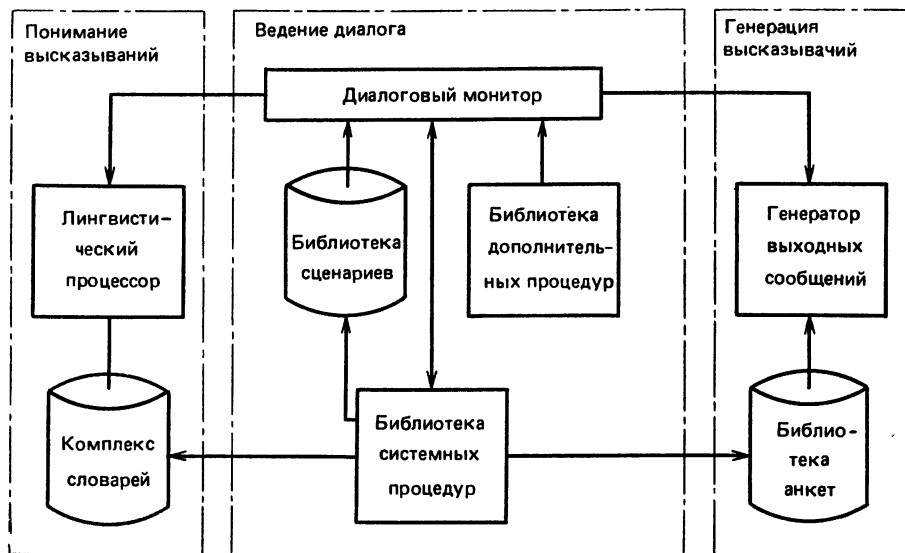


Рис. 3.1. Структура системы АИСТ

АИСТ. В случае необходимости (например, выбор другой типовой задачи, устранение непониманий и т. п.) пользователь может прервать текущий сценарий с помощью специальных команд перехвата инициативы, которые могут вводиться в моменты, когда АИСТ находится в ожидании очередного высказывания пользователя.

Ядро компонента ведения диалога образуют диалоговый монитор и библиотека сценариев. Диалоговый монитор фактически является интерпретатором правил, составляющих сценарии решения типовых задач. Кроме того, в его функции входит обработка перехватов инициативы, а также формирование и поддержка фокуса внимания (т. е. совокупности параметров конкретной решаемой задачи). В библиотеке сценариев содержатся частично упорядоченные множества правил, задающих тематическую и локальную (за исключением перехватов инициативы пользователем) структуры диалога (см. п. 1.3.1) при решении типовых задач. Правила являются описаниями шагов диалога при решении тех или иных типовых задач. Каждое правило определяет: экранный формат, используемый на данном шаге для вывода высказывания системы и ввода высказывания пользователя; параметры решаемой задачи, связываемые с окнами экранного формата; ожидаемые ответы пользователя и соответствующие этим ответам действия (т. е. обращения к компонентам понимания или генерации высказываний или к программам, входящим в состав библиотек системных и дополнительных процедур); выбор правила, описывающего следующий шаг диалога.

Библиотеку системных процедур составляют прикладные программы, выполняющие обработку данных на различных этапах решения типовых задач. В частности, с помощью этих программ осуществляется ведение комплекса словарей, библиотеки сценариев, формирование, каталогизация и поиск анкет в библиотеке анкет и т. п.

Библиотека дополнительных процедур состоит из программ, осуществляющих подготовку параметров и вызов внешних программ (например, модуля связи с СУБД), которые не входят в систему АИСТ, но могут использоваться для решения каких-то подзадач.

Компонент понимания высказываний обеспечивает перевод анкет, заполненных на русском языке, во внутреннее представление. При переводе учитываются как тип анкеты (она может быть предназначена либо для ввода, либо для поиска и выдачи данных), так и способы представления в БД фактографических и содержательных реквизитов. Компонент понимания высказываний состоит из проблемно-независимого лингвистического процессора [Преображенский, 1982], ориентированного на анкетную форму представления входной информации, и комплекса словарей, содержащего все необходимые для перевода сведения о языке, предметной области и структуре БД.

Процесс анализа входных высказываний включает три этапа: структурный анализ; анализ полей анкеты; формирование внутреннего представления. На первом этапе определяется тип анкеты (документ или запрос) и выделяются отдельные пары «имя реквизита — выражение», соответствующие полям анкеты. На втором этапе имена реквизитов и выражения (значения реквизитов) переводятся с русского языка на промежуточный, который отличается от языка внутреннего представления тем, что в нем не учитываются связи между файлами БД. Практически на данном этапе имена реквизитов переводятся в коды атрибутов БД, а выражения — в термины, используемые в БД для представления значений соответствующих атрибутов. Для этого используются морфологический анализ, элементы синтаксического и семантического анализа, вспомогательный словарь, классификаторы, тезаурус, а также информация о свойствах атрибутов БД, содержащаяся в словаре имен.

Вспомогательный словарь содержит морфологическую информацию о словах и словосочетаниях русского языка, известных системе (т. е. о ключевых словах, имеющихся в классификаторах, тезаурусе и словаре имен). Классификаторы включают перечни всех возможных для использования нормализованных значений фактографических реквизитов и существующие между ними отношения (синонимия, род-вид и т. д.). В тезаурус входит множество ключевых слов, разбитое на классы условной эквивалентности, между которыми установлены родовидовые и ассоциативные отношения. Названия классов — дескрипторы используются в БД для внутреннего представления значений содержательных реквизитов. Словарь имен представляет собой перечень имен реквизитов, соответствующих названиям (кодам) атрибутов и других элементов логической структуры БД. Для каждого атрибута в словаре имен указаны его свойства (тип и длина значения, возможность выполнения поиска по данному атрибуту и т. п.), а также имена файлов, в структуре которых определен этот атрибут.

Третий этап анализа входных высказываний заключается в объединении результатов перевода отдельных пар «имя реквизита — выражение» в единое внутреннее представление, построенное с учетом типа анкеты (документ или запрос) и логических связей между файлами БД. Принципиальное отличие формирования внутреннего представления запроса от документа состоит в том, что в общем случае запрос может содержать довольно сложные операции, определяющие условия поиска и вычисления, отсутствующие при обработке документов. Языком внутреннего представления является язык ДИОД системы ДИСОД.

Последний функциональный компонент системы АИСТ — компонент генерации высказываний служит для формирования и выдачи на экран дисплея вопросов, обращенных к пользователю, и сообщений о результатах решения тех или иных задач. Вопросы подразделяются на простые, альтернативные и вопросы с фиксированной структурой ответа. Простые вопросы генерируются в тех случаях, когда диалоговому монитору известна требуемая для решения текущей подзадачи прикладная программа и нужно получить недостающие параметры этой подзадачи. Например: «Введите дату в формате чч.мм.гг» или «Укажите номер в формате ХХХ». Альтернативные вопросы предоставляют пользователю возможность выбора одной или нескольких сущностей (задач, параметров задач и т. п.) из перечня, выдаваемого на экран дисплея. В первом случае (выбор одной сущности) перечень называется меню, а во втором (выбор нескольких сущностей) — каталогом. Вопросы с фиксированной структурой ответа — это анкеты. Анкета представляет собой совокупность полей, каждое из которых за-

нимает одну или несколько строк. В начале поля указано имя реквизита, а оставшаяся часть служит для ввода пользователем выражения (фактографического или содержательного), представляющего собой значение реквизита.

Сообщения, генерируемые системой АИСТ, подразделяются на диагностические, сообщения типа «помощь» и отчеты. Диагностические сообщения чаще всего используются для указания на неправильные действия пользователя (например, в анкете не заполнено ни одно поле), а также для выдачи сведений, информирующих пользователя о завершении каких-то этапов решения задачи. Сообщения типа «помощь» содержат разъяснения возможностей, предоставляемых системой на каждом этапе решения задачи, и рекомендации относительно их использования в различных ситуациях. Отчеты используются для представления документов, полученных в результате поиска в БД. В общем случае отчет имеет форму таблицы, снабженной заголовком, номерами страниц, шапками (именами столбцов), боковиками (именами строк) и т. п.

Генерация всех типов высказываний в системе АИСТ осуществляется на основе заранее подготовленных шаблонов (форм), которые могут содержать переменные для подстановки конкретных значений. Шаблоны подразделяются на два класса: статические, заданные в сценарии диалога (к ним относятся диагностические сообщения, сообщения типа «помощь», простые вопросы и вопросы типа «меню»), и динамические, формируемые в процессе диалога либо в соответствии с информационной потребностью пользователя (анкеты, отчеты), либо в соответствии с конкретной ситуацией, возникшей при решении задачи (вопросы с использованием каталогов).

3.1.2. Представление знаний. В соответствии с классификацией знаний ЕЯ-систем (см. § 1.2) рассмотрим знания системы АИСТ относительно следующих областей: 1) проблемная область; 2) область языка и диалога; 3) область пользователя; 4) область системы. Знания о проблемной области представлены в АИСТ в основном декларативно, что упрощает процесс настройки системы на различные проблемные области. Они включают: знания о решаемых задачах (библиотека сценариев); знания о сущностях предметной области и об отношениях между ними (классификаторы, тезаурус); знания о структуре БД и свойствах атрибутов (словарь имен). Все указанные наборы данных реализованы в виде технологических файлов БД ДИСОД, что позволяет использовать, во-первых, стандартные средства СУБД для их сопровождения (инициализации, восстановления, получения страховых копий и т. п.) и, во-вторых, средства АИСТ для корректировки этих файлов при настройке на новые проблемные области.

Знания о языке и диалоге подразделяются на декларативные и процедурные. К декларативным относятся: лексические знания (вспомогательный словарь); знания о статических форматах и шаблонах высказываний системы (библиотека сценариев); знания о динамических шаблонах высказываний системы (библиотека анкет); знания о структуре диалога (библиотека сценариев). Вспомогательный словарь и библиотека анкет реализованы также в виде технологических файлов БД ДИСОД. Процедурное представление используется в системе АИСТ для тех видов знаний, которые не зависят (или слабо зависят) от специфики проблемной области. Это позволяет, не уменьшая гибкости, повысить эффективность работы системы. К процедурным знаниям о языке и диалоге относятся знания о морфологии (парадигмах слов русского языка), синтаксисе (правилах образования словосочетаний и однородных конструкций) и семантике русского языка, выраженные в соответствующих программах лингвистического процессора, и знания о способах обработки перехватов инициативы, содержащиеся в программах диалогового монитора. Кроме того, к процедурным знаниям относятся знания о способах решения типовых подзадач и способах вызова внешних программ, представленные соответственно прикладными программами из библиотеки системных процедур и дополнительными процедурами.

К знаниям о пользователе относятся лишь содержащиеся в библиотеке анкет сведения о принадлежности той или иной анкеты конкретному пользователю или группе пользователей.

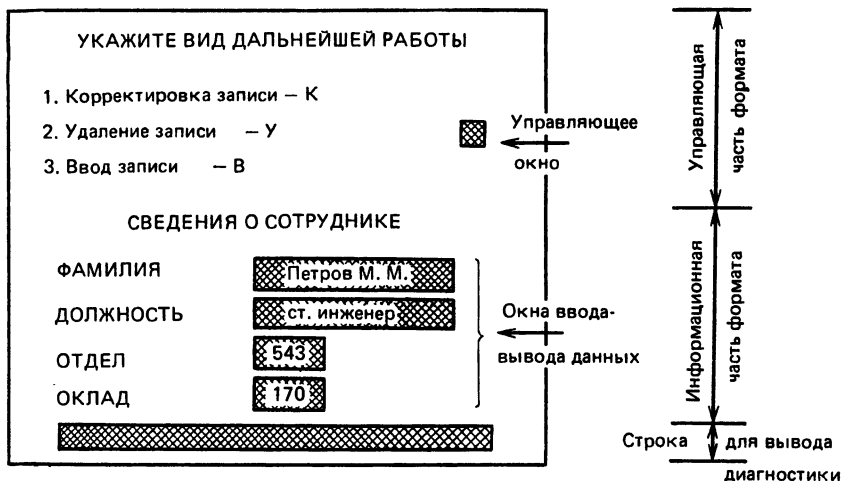


Рис. 3.2. Структура экранного формата

В отличие от рассмотренных выше знаний, которые по уровню представления (см. § 1.2) являются знаниями нулевого уровня, знания АИСТ о системе (т. е. о себе) являются метазнаниями. Это знания о способах представления всех декларативных знаний, используемых в системе АИСТ.

3.1.3. Языковые и диалоговые возможности. Взаимодействие пользователя с системой АИСТ осуществляется с помощью экранных форматов, которые в общем случае могут состоять из двух частей: управляющей и информационной (рис. 3.2). Управляющая часть информирует пользователя о возможных действиях на данном шаге диалога и позволяет выбрать одно из указанных действий. Обычно управляющая часть имеет форму меню и содержит окно для ввода метки выбранного пользователем действия. Помимо описаний действий, направленных на дальнейшее решение задачи, в меню могут включаться альтернативы, обеспечивающие выдачу разнообразных подсказок, которые могут быть полезны пользователю в случае затруднений при заполнении информационной части.

Информационная часть формата служит для ввода и вывода участниками общения параметров решаемой задачи. Последняя строка информационной части формата используется для вывода диагностических сообщений.

По своему назначению языковые средства системы АИСТ подразделяются на две группы: средства решения задач; средства описания диалоговых приложений. К средствам решения задач относятся языки управления диалогом и выражения информационных потребностей. Первый из них служит пользователю для управления диалогом в соответствии с текущим состоянием решаемой задачи и преследуемыми целями. Данный язык составляют команды управления диалогом и перехвата инициативы.

Команды управления диалогом позволяют на каждом шаге диалога выбрать то из возможных продолжений, которое соответствует текущим целям пользователя. Для выбора продолжения диалога могут использоваться два способа. Первый — это ввод команды в специально отведенное для этого окно экранного формата. Команда в данном случае представляет собой метку одной из альтернатив, указанных в меню управляющей части формата. Второй способ состоит в том, что выбираемое продолжение диалога отмечается специальным символом «*», предоставляемым в окно, расположенное рядом с описанием этого продолжения в меню.

Команды перехвата инициативы служат для выполнения стандартных управляющих воздействий и могут применяться на любом шаге диалога независимо от сценария. Обычно эти команды выводятся на функциональные клавиши терминала. По характеру управляющих воздействий они подразделяются на три группы: команды отмены (текущего шага или текущего диалога); команды перехода (к альтернативному диалогу, т. е. к решению другой задачи или к той же задаче, но с другими параметрами) и возврата к текущему состоянию прерванного диалога; команды обслуживания («помощь», «листание», «получение твердой копии» и т. п.).

Язык выражения информационных потребностей служит для ввода пользователем данных в окна информационной части экранных форматов в ответ на простые вопросы и вопросы, использующие анкеты. (Ответы на вопросы, использующие каталог, вводятся так же, как и ответы на вопросы в форме меню.) Ответами на эти вопросы служат фактографические или содержательные выражения (см. п. 3.1.1), представленные в виде одного или нескольких предложений русского языка. Ниже приводятся правила построения выражений.

Содержательные выражения представляют собой текст на русском языке. Последним элементом каждого предложения текста является точка, а предложения отделяют друг от друга пробелом. За каждым знаком препинания внутри предложений следует пробел. Точки, ограничивающие инициалы и общепринятые сокращения, могут не отделяться пробелом от следующих за ними слов. В анкетах поиска в содержательных выражениях могут использоваться функциональные слова В1, Н1, Н2 и А1. Они предназначены для более точного выражения информационной потребности пользователя. Функциональное слово, поставленное перед словом или словосочетанием, вызывает автоматическое включение в запрос к БД дополнительных терминов внутреннего представления (дескрипторов), находящихся с данным термином в определенных отношениях (В1 позволяет включить родовой дескриптор, Н1 — видовые, Н2 — иерархию видовых дескрипторов и А1 — дескрипторы, находящиеся с данным в ассоциативном, т. е. в любом, кроме видового и родового, отношении).

Фактографические выражения представляют собой совокупности значений соответствующих реквизитов, выраженных в виде слов и словосочетаний русского языка и связанных фиксированными отношениями (и, или, от ... до, меньше, больше, кроме, раньше, позже и т. п.)

Языковые средства описания диалоговых приложений служат для ввода в библиотеку сценариев правил, с помощью которых выражаются знания о диалоге, направленном на решение типовых задач пользователя, т. е. о возможных способах разбиения этих задач на подзадачи, методах решения отдельных подзадач и об экранных форматах и статических шаблонах, используемых для ввода и вывода высказываний пользователя и системы на различных шагах диалога. Язык описания диалоговых приложений имеет анкетную форму [Дракин и др., 1988], облегчающую и в какой-то степени автоматизирующую процесс ввода или модификации знаний АИСТ о диалоге, направленном на решение типовых задач.

3.1.4. Взаимодействие пользователя с системой АИСТ. С помощью системы АИСТ могут быть реализованы диалоговые приложения (типовые задачи), позволяющие удовлетворять разнообразные информационные потребности пользователя. Ниже рассматривается пример поиска информации в БД. Удовлетворение информационных потребностей пользователя осуществляется в данном случае средствами диалоговых приложений АНКЕТА, ПОИСК и ТЕРМИН, которые реализуют соответственно следующие этапы глобальной структуры диалога: 1) формирование анкет в соответствии с информационной потребностью пользователя; 2) итерационный поиск информации в БД на основе использования сформированных анкет; 3) уточнение причин неудач и уточнение терминов, применяемых для заполнения анкет в ходе итерационного поиска. Тематическая и локальная структура диалога при выполнении указанных этапов полностью (за исключением перехватов инициативы) определяется сценариями типовых задач АНКЕТА, ПОИСК и ТЕРМИН соответственно. Поэтому наряду с термином «типичная

Проставьте символ * около имен реквизитов,
которые будут включены в анкету поиска

- * Дата прибытия
- Дата отъезда
- * Фамилия
- * Инициалы
- Специальность
- * Страна
- Фирма
- * Цель командировки

Рис. 3.3. Пример формирования анкеты поиска

задача» далее будет использоваться в качестве синонима термин «типовой диалог».

В сценариях типовых задач АНКЕТА, ПОИСК и ТЕРМИН предусмотрены действия (подзадачи) по преодолению следующих видов локальных неудач: 1) попытка использовать несуществующую анкету; 2) попытка создать анкету, включающую реквизиты, к которым пользователь не допущен или которые отсутствуют в БД; 3) применение в ответах на вопросы системы (в том числе и при заполнении анкет) слов, которые неизвестны системе; 4) получение неточных или неполных результатов поиска.

Инициатива во всех типовых диалогах принадлежит системе. Однако пользователь с помощью соответствующих команд (см. п. 3.1.3) может перехватить инициативу для получения помощи, прерывания решения текущей типовой задачи и перехода к решению другой (альтернативной) типовой задачи и т. п.

После инициации системы АИСТ на экран дисплея выдается так называемый ключевой формат, содержащий меню типовых задач, доступных пользователю. Пусть выбрана задача ПОИСК. Тогда на следующем шаге АИСТ требует ввести имя анкеты поиска. Если пользователь не имеет собственной анкеты поиска или его не устраивают те, которые уже занесены в библиотеку анкет, то для ее создания (а также для просмотра анкет, имеющихся в библиотеке) следует перейти к задаче АНКЕТА. Это осуществляется с помощью команды перехода к альтернативной задаче (функциональный клавиш ПФ1). После ввода этой команды состояние решения задачи ПОИСК запоминается, а на экран дисплея снова выдается ключевой формат, в подсвеченное окно которого вводится имя: АНКЕТА.

На первом шаге диалога АНКЕТА пользователю предлагается меню для выбора вида работы. Основными видами работы являются просмотр, поиск, редактировка, удаление и создание анкет. Пусть выбирается формирование анкеты поиска. После этого АИСТ просит пользователя ввести имя интересующей его предметно-тематической области БД. Список областей может быть выдан в виде меню. Пусть пользователь выбрал предметно-тематическую область «Сведения о заграничных командировках сотрудников». После этого на следующем шаге в управляющей части формата предлагаются такие альтернативы: 1) ввести список имен реквизитов, которые должны войти в создаваемую анкету; 2) воспользоваться подсказкой. Имена реквизитов должны вводиться в подсвеченное окно, составляющее информационную часть формата. При выборе второй альтернативы в информационную часть формата выводится каталог возможных (в рамках данной предметно-тематической области) имен реквизитов. Для включения в анкету реквизита, соответствующего информационной потребности пользователя, слева от него достаточно проставить символ «*», как это показано на рис. 3.3. На следующем шаге от пользователя требуется указать имя созданной анкеты, под которым она будет внесена в библиотеку анкет. Аналогично может быть создана и анкета выдачи.

Введите условия поиска:

Дата прибытия: раньше 02.11.86 _____

Фамилия: _____

Инициалы: _____

Страна: Бенилюкс _____

Цель командировки: заказ оборудования для горных предприятий _____

Рис. 3.4. Пример заполнения анкеты поиска

После завершения процесса создания анкет пользователь возвращает в запомненное состояние прерванный диалог по решению задачи ПОИСК. Для этого применяется соответствующая команда (она реализована функциональным клавишем ПФ4). В восстановленном состоянии от пользователя по-прежнему требуется указать имя анкеты поиска. Он вводит имя созданной анкеты, и на следующем шаге в информационную часть формата выводится только что сформированная анкета. Заполняя поля этой анкеты, например, так, как показано на рис. 3.4, пользователь формулирует условия поиска информации в базе данных.

Пусть при обработке заполненной анкеты система обнаруживает, что слово «Бенилюкс», присутствующее в поле Страна, ей неизвестно. Тогда на следующем шаге система предлагает такие альтернативы: вернуться на предыдущий шаг с тем, чтобы заменить неопознанное слово другим словом; продолжить обработку без учета этого слова; определить неопознанное слово через известные системе понятия. Наиболее интересной является третья альтернатива, выбор которой инициирует уточняющий поддиалог. При ее выборе определение неопознанного термина может происходить либо через синонимы, либо через видовые понятия. Пусть выбирается второй способ. Тогда в соответствующее окно информационной части формата пользователь вводит: «Бельгия, Нидерланды, Люксембург». Если при определении терминов у пользователя возникают затруднения, то он может прервать задачу ПОИСК, перейти к задаче ТЕРМИН, просмотреть допустимые лексические единицы и снова вернуться в то же самое состояние прерванного диалога.

На следующем шаге диалога на экран выдается сообщение о результатах поиска по условию, сформированному на предыдущих шагах, например НАЙДЕНО 15 ЗАПИСЕЙ. На этом шаге предлагаются следующие альтернативы: указать имя анкеты выдачи; получить список анкет выдачи, связанных с данной анкетой поиска; использовать различные виды стратегий поиска; осуществить возврат к шагу заполнения анкеты, чтобы скорректировать условия поиска. Пусть выбрана четвертая альтернатива. Тогда выводится заполненная анкета и пользователь проставляет указатель расширения поиска в поле Цель командировки. Например, его значением становится «заказ оборудования для Н1 горных предприятий». В этом случае в запрос к БД будут включены (со связкой ИЛИ) дескрипторы, связанные отношением вид-род с дескриптором «горнодобывающие предприятия» (предполагается, что «горные предприятия» — один из его синонимов). Обычно это приводит к увеличению числа найденных записей. Такой процесс имеет итеративный характер. Здесь могут также использоваться и автоматические стратегии поиска.

После завершения итеративного поиска пользователь указывает имя анкеты выдачи (либо введя имя в соответствующее окно формата, либо сделав отметку в выведенном по его требованию каталоге связанных анкет). Анкета выводится в информационную часть формата на следующем шаге диалога. Для включения в отчет значений, соответствующих тем или иным полям анкеты выдачи, достаточно проставить слева от них символ * (рис. 3.5). Выдаваемый отчет имеет

Проставьте символ * около имен реквизитов,
которые будут включены в отчет

Анкета выд102

- * Фамилия
- * Инициалы
- * Страна
- Фирма
- * Цель командировки

Рис. 3.5. Пример формирования анкеты выдачи

стандартную структуру — каждому отмеченному полю анкеты выдачи соответствует столбец значений, снабженный заголовком (рис. 3.6). С помощью системных команд отчет можно пролистать, переслать на другой видеотерминал или вывести на устройство печати. После этого пользователю предлагаются следующие возможности: 1) завершить сеанс взаимодействия; 2) вернуться к шагу выбора анкеты поиска; 3) вернуться к шагу задания (или изменения) условий поиска; 4) вернуться к шагу выбора анкеты выдачи; 5) вернуться к шагу задания списка выдаваемых значений.

3.1.5. Возможности настройки и адаптации. Данные возможности в системе АИСТ основаны на широком использовании декларативных знаний для выражения специфики проблемной области и индивидуальных потребностей пользователей (см. п. 3.1.2). Настройка АИСТ может осуществляться по следующим параметрам: лексика языка общения; семантика (область интерпретации) языка общения; логическая структура БД; типовые задачи и структура диалога, направленного на их решение. Адаптация к пользователю реализуется средствами каталогизации анкет, отражающих индивидуальные информационные потребности, и средствами включения в язык общения персональной лексики пользователя (см. пример из п. 3.1.4).

Простота и технологичность настройки и адаптации АИСТ являются следствием использования единого способа хранения знаний, зависящих от специфики проблемной области пользователей, в виде технологических файлов БД ДИСОД. Хотя это и приводит к некоторому ухудшению временных характеристик системы, но вместе с тем позволяет: хранить значительные объемы знаний; применять единую технологию и средства сопровождения как данных о проблемной области (информационных файлов БД), так и знаний, используемых АИСТ; реализовать с помощью АИСТ типовые диалоги, предназначенные для ввода и корректировки проблемно-ориентированных и ориентированных на пользователя знаний.

Найдено 132 записи
Ширина отчета 80

Лист 1 22.11.85

Фамилия	Инициалы	Страна	Цель командировки
Ларионов	С. Л.	Бельгия	Заказ оборудования для карьера "Верхнеардынский"
Шелетов	И. В.	Нидерланды	Заказ оборудования для промывки золота
Шибанова	И. А.	Бельгия	Заказ оборудования для карьера "Нижекозельский"

Рис. 3.6. Пример отчета

Процесс настройки АИСТ включает два этапа: создание и первоначальная загрузка технологических файлов в отсроченном (пакетном) режиме при разворачивании системы; пополнение и корректировка технологических файлов в диалоговом режиме в процессе эксплуатации системы.

Создание технологических файлов осуществляется средствами, предоставляемыми СУБД. В состав технологических файлов входят: вспомогательный словарь (ВС), классификатор (КЛ), тезаурус (ТЗ), словарь имен (СИ), библиотека сценариев (БС) и библиотека анкет (БА). В БС и ВС соответственно загружаются поставляемые в составе АИСТ сценарии типовых диалогов и «нуль-словарь», который содержит около 8000 основ общеупотребительных слов русского языка со всей необходимой грамматической информацией. Нуль-словарь и сценарии типовых диалогов могут рассматриваться как «врожденные» знания АИСТ. В КЛ и ТЗ загружаются соответственно классификаторы и информационно-поисковый тезаурус, описывающие различные сущности проблемной области и отношения между ними, а в СИ — сведения о логической структуре БД и свойствах атрибутов (эти сведения частично «перекачиваются» из словаря-справочника данных СУБД ДИСОД).

Последняя операция по первоначальной загрузке файлов заключается в наполнении ВС лексикой ТЗ, КЛ и СИ. Для этого используются специальные программные средства, входящие в состав АИСТ, которые обеспечивают автоматическое формирование в ВС словарных статей словосочетаний и автоматизированный ввод в ВС основ (вместе с грамматической информацией) слов, отсутствующих в ВС, но входящих в словосочетания, представленные в ТЗ, КЛ и СИ.

Необходимость процесса пополнения и корректировки технологических файлов (т. е. второго этапа настройки) связана с непредусмотренными на первом этапе изменениями информационных потребностей пользователей, проблемной области и логической структуры БД. Средства, используемые на втором этапе, могут быть разделены на общие и специальные. Общие средства реализованы в виде типового диалога (типовой задачи) ФАЙЛЫ, позволяющего вводить, корректировать и удалять записи любого из технологических файлов. Специальные средства учитывают специфику тех или иных технологических файлов и предоставляют пользователю дополнительные удобства. Эти средства реализованы в виде двух типовых диалогов: ТЕРМИН и ФОРМА.

Диалог ТЕРМИН позволяет просматривать, вводить и корректировать лексику, используемую в ответах на вопросы системы. Обращение к этому диалогу обычно происходит при локальных неудачах, вызванных появлением в ответах пользователя незнакомых слов. В результате уточняющего диалога (см. п. 3.1.4) может потребоваться ввод дополнительных синонимов или новых дескрипторов и терминов классификатора. В рамках диалога ТЕРМИН осуществляется корректировка ТЗ и КЛ, а также соответствующая корректировка ВС. При этом от пользователя не требуется никаких лингвистических знаний, так как АИСТ сам пытается определить основы (и грамматическую информацию) вводимого слова или словосочетания, а в случае получения неоднозначных результатов производит синтез альтернативных словоформ и просит указать правильную словоформу.

Типовой диалог ФОРМА служит для корректировки существующих сценариев типовых задач или ввода сценариев новых типовых задач. Типовая задача ФОРМА предусматривает корректировку или ввод следующих элементов сценария: 1) имени типовой задачи; 2) описаний параметров типовой задачи; 3) описаний экранных форматов и статических шаблонов, используемых для ввода-вывода высказываний при решении типовой задачи (т. е. в ходе типового диалога); 4) содержания экранов, выдаваемых по команде «помощь»; 5) описаний шагов типового диалога. В отличие от типовых задач, рассмотренных ранее, задача ФОРМА требует от пользователя специальных знаний. Кроме того, следует подчеркнуть, что возможности АИСТ по обработке данных (т. е. решению подзадач, возникающих в типовых диалогах) ограничены имеющимися в библиотеке системных процедур прикладными программами (см. п. 3.1.1). Поэтому в случае возникновения подзадач, которые не могут быть решены этими при-

кладными программами, следует включить в библиотеку дополнительных процедур программы обращения к внешним прикладным программам, обеспечивающим обработку данных в соответствии с требованиями создаваемого типового диалога.

3.1.6. Реализация системы. Система АИСТ реализована на языке Ассемблер ЕС ЭВМ и может использоваться на машинах с емкостью оперативной памяти не менее 2 Мбайт и операционной системой ОС 6,1 и выше. Требуемая емкость оперативной памяти при работе в пакетном режиме (в этом случае используется только лингвистический процессор) не более 300 Кбайт, а в диалоговом режиме (используются все компоненты) определяется по формуле $(180 + 45n)$ Кбайт, где n — число одновременно обслуживаемых терминалов. Работа в многотерминальном режиме осуществляется под управлением системы типа КАМА.

Время выполнения шага диалога без учета времени на преобразование значений содержательных реквизитов составляет от 0,5 до 2 с. Преобразование значений содержательных реквизитов во внутреннюю форму представления при объеме вспомогательного словаря 70 000 словарных статей составляет в среднем от 0,1 до 0,2 с на слово.

Документация на систему АИСТ выполнена в соответствии с требованиями ЕСКД. В состав эксплуатационной документации входят: техническое описание; руководство системному программисту; руководство по подготовке данных; руководство оператору; руководство пользователю; структура данных; таблица программ на магнитной ленте. Магнитная лента содержит следующие файлы: библиотеку символьных модулей; библиотеку макрокоманд; библиотеку заданий; библиотеку процедур; нуль-словарь; библиотеку сценариев.

Система АИСТ передана в СФАП Алгоритм в 1986 г. При заказе на нее следует ссылаться на «Изделие АИСТ АТЛЗ.060.302».

3.2. Коммерческая система ЛИНГВИСТ

А. Ф. Киселев, Э. В. Попов

3.2.1. Общее описание. Система предназначена для создания на персональных ЭВМ (ПЭВМ) информационно-поисковых систем (персональных баз данных), ориентированных на непрограммирующих конечных пользователей. Информационно-поисковая система для конкретного пользователя создается путем настройки системы ЛИНГВИСТ на данную проблемную область и может быть применена для организации рабочего места конечного пользователя с целью автоматизации его деятельности, связанной с переработкой информации, и удовлетворения его информационной потребности. Обычно в такого рода деятельности пользователь имеет дело с разнотипной информацией довольно сложной структуры, представляемой в виде различных документов. Например, медицинский документ «История болезни» содержит фамилию больного, дату рождения, место рождения и т. п. (фактографические реквизиты), диагноз, описание протекания заболевания и т. п. (содержательные реквизиты), различные результаты обследований в виде изображений — графиков и диаграмм (графические реквизиты). Разница между фактографическими и содержательными реквизитами заключается в том, что фактографические реквизиты принимают значения из определенного, как правило, заранее классифицированного перечня понятий, а значениями содержательных реквизитов являются произвольные тексты русского языка.

Система ЛИНГВИСТ организует работу пользователя с документами на ПЭВМ. Она обеспечивает: 1) ввод документов (данных) в базу данных системы; 2) хранение документов в виде взаимосвязанных файлов БД; 3) поиск документов по текстовым (фактографическим и содержательным) и графическим реквизитам документов; 4) отображение на дисплее и бумажном носителе найденных документов.

Успех в удовлетворении информационной потребности пользователя во многом определяется естественностью общения с системой и отображения хранимых документов. Процесс общения ведется в соответствии с типовыми или настраиваемыми на конкретного пользователя сценариями. Отображение документов и запросов осуществляется в различной форме в зависимости от подготовленности пользователя и его привычек работать с информацией в том или ином виде.

Для взаимодействия с пользователем система ЛИНГВИСТ предоставляет следующие возможности: 1) неструктурированный естественный русский язык (далее по тексту ЕЯ) для задания нестандартных запросов по фактографическим реквизитам; 2) язык таблиц (естественно-языковых динамических шаблонов) для формирования пользователем нестандартных запросов по фактографическим реквизитам и отображения документов в виде таблиц; 3) язык анкет (заранее определенных естественно-языковых статических шаблонов) для реализации ввода, отображения документов и формирования стандартных запросов к БД. Отдельные поля анкеты соответствуют текстовым или графическим реквизитам. ЛИНГВИСТ позволяет отображать одни и те же данные, хранимые в БД, в различном сочетании и получать производные данные, что дает возможность формировать новые документы из существующих.

Одним из основных процессов при работе с данными является поиск информации (документов). Поиск документов по фактографическим реквизитам осуществляется на полное или частичное совпадение значения реквизита в запросе и документе, кроме того, для чисел и дат — на больше, меньше и на интервал. При поиске по содержательным реквизитам ЛИНГВИСТ производит индексирование текста, т. е. автоматически выделяет в тексте ключевые понятия и по ним обеспечивает поиск. ЛИНГВИСТ позволяет «индексировать» и изображения, благодаря чему возможен поиск по графическим реквизитам документов.

Система ЛИНГВИСТ обучает пользователя и оказывает ему помощь в работе. Она информирует пользователя о своих возможностях, структуре хранимых документов, информационном обеспечении, используемом системой, и способах работы на каждом шаге диалога. Это дает возможность пользователю легко приобретать навыки работы с системой и получать информацию не только о хранимых документах, но и об устройстве системы и о ее знаниях.

Общая укрупненная схема системы ЛИНГВИСТ приведена на рис. 3.7. Основные программные компоненты изображены в виде прямоугольников, промежуточные результаты функционирования программных компонентов — в виде овалов, основные информационные компоненты (знания системы) — в виде цилиндров. Основные информационные компоненты и способы их представления описаны в п. 3.2.2. Основными программными компонентами системы ЛИНГВИСТ являются: диалоговый компонент; обработчик ЕЯ-запросов; табличный интерфейс; анкетный интерфейс; графический интерфейс; интерпретатор моделей; система управления базой данных; компонент настройки на проблемную область.

Диалоговый компонент поддерживает взаимодействие с пользователем в соответствии со сценариями диалогов. В системе заложен ряд типовых сценариев для работы с БД пользователя и для проведения стандартных работ технологического характера. Пользователь может определить собственные сценарии диалогов и переопределять их в ходе работы. Диалоговый компонент работает в поликономном режиме, что позволяет, например, одновременно просматривать документы пользователя и логическую структуру БД.

Обработчик ЕЯ-запросов осуществляет обработку естественно-языковых запросов на поиск информации по фактографическим реквизитам. В основу функционирования обработчика ЕЯ-запросов положен прагматический подход [Файн, 1985]. Отличительная особенность прагматического подхода к анализу естественного языка состоит в следующем: вместо анализа ЕЯ-текста по полной схеме (см. § 1.2) анализировать текст только в той степени, которая необходима, чтобы получить адекватную реакцию системы.

В системе ЛИНГВИСТ под адекватной реакцией системы понимается способность проинтерпретировать ЕЯ-запрос в терминах представления данных пользователя (подсхемы пользователя) и запустить процедуры извлечения информа-

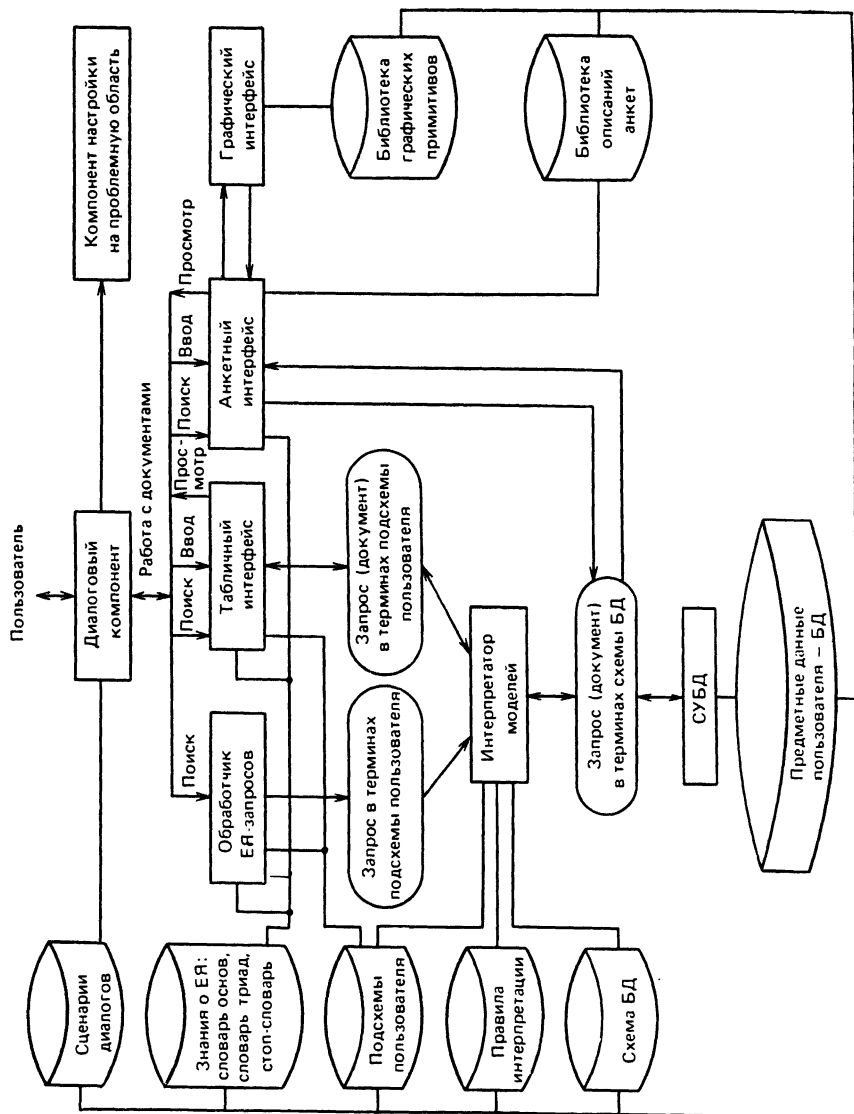


Рис. 3.7. Общая схема системы ЛИНГВИСТ

Перевозка	Груз	Транспорт	Объем	Год
	Бурый уголь	Железнодорожный транспорт	?	1987

Рис. 3.8. Пример табличного запроса

ции из БД системы. Предполагается, что всякое обращение к системе является заданием, которое она должна выполнить, введя в действие соответствующие процедуры. Такой подход не требует разработки всеобъемлющего анализатора текстов русского языка для получения полного «смысла» входного запроса, эффективную реализацию которого сложно создать даже на больших ЭВМ. ЛИНГВИСТ производит полный морфологический анализ, выделяет в запросе информативные слова и словосочетания и определяет по словарю основ атрибуты БД, значениями которых могут являться выделенные слова и словосочетания (при этом учитываются падежи слов, одушевленность и другая информация). ЛИНГВИСТ информирует пользователя, как система «поняла» его ЕЯ-запрос, а в случае невозможности построить однозначную интерпретацию входного запроса предоставляет множество возможных интерпретаций в виде меню и предлагает выбрать приемлемую. В качестве примера приведем следующий ЕЯ-запрос: «Сколько бурого угля перевезено в 1987 году?» ЛИНГВИСТ может обрабатывать неполные и даже синтаксически неверные запросы. Так, система проинтерпретирует (возможно, неоднозначно) приведенный выше запрос, если он будет сформулирован в следующем виде: «Бурого угля в 1987 году?» С нашей точки зрения, обработка таких запросов очень важна в реальных системах и просто незаменима при построении систем, работающих с голоса (предполагается подключение к системе ЛИНГВИСТ анализатора речи).

Табличный интерфейс осуществляет отображение реквизитов документов в виде таблиц. Каждой таблице соответствует часть (отношение) подсистемы пользователя. Табличный интерфейс позволяет строить новые таблицы, возможно, на основании уже существующих и запоминать их, а также конструировать из существующих таблиц запросы на поиск информации. Пример табличного запроса приведен на рис. 3.8.

Анкетный интерфейс позволяет осуществлять ввод, поиск, корректировку и отображение данных из БД, используя при этом заранее построенные шаблоны — анкеты. В отличие от отработчика ЕЯ-запросов и табличного интерфейса анкетный интерфейс позволяет работать с любыми реквизитами документов: фактографическими, содержательными и графическими. В анкетный интерфейс заложены типовые сценарии для ввода, поиска и корректировки документов, которые могут быть изменены и дополнены пользователем. Пользователем должны быть построены следующие анкеты: для ввода документов, для их поиска (включает только поисковые реквизиты документа), для просмотра и корректировки документов, найденных в результате поиска (может совпадать с анкетой для ввода), и для просмотра основных реквизитов найденных документов в виде списка, в котором под значения реквизитов каждого документа отводится одна или несколько строк на дисплее (последняя анкета является необязательной). Можно осуществлять связывание анкет, отображающих одни и те же или различные документы: при вводе значения реквизитов связи из вводимого документа автоматически переносятся в связанный документ, при просмотре найденных документов можно перейти на просмотр связанных документов. При этом запрос на поиск связанных документов формируется автоматически из значений реквизитов связи. Все анкеты строятся пользователем в диалоге с системой в режиме непосредственного редактирования анкет. На рис. 3.9 показан пример анкеты ввода.

Графический интерфейс позволяет осуществить ввод в БД, поиск и отображение изображений, встречающихся в документах. Изображения строятся из

УДК 331.97.681.142.7 _____

Номер карточки 19_____

Тематика Вычислительные сети

Автор Гафт Л. Ш., Зарубин И. И. _____

Заглавие Вопросы экономического и организационного обеспечения функционирования сетей вычислительных центров _____

Название документа Организационно-экономические аспекты использования _____
вычислительной техники _____

Сведения о документе

Том _____ номер _____ стр. 3 — 10 _____ объем 8 _____

Издательство Институт кибернетики им. В. М. Глушкова _____

Место издания Киев _____ Год издания 1986 _____

Аннотация Изложены положения подхода к проблеме создания сетей ВЦ в —

отраслях народного хозяйства и отдельных регионах; сформулированы основные —
направления работ по развитию локальных вычислительных сетей

Рис. 3.9. Пример анкеты ввода

графических объектов, определенных пользователем, объекты — из заложенных в систему графических примитивов (точка, прямая, линия, круг, параллельные линии и т. п.) и ранее построенных объектов. Система способна отыскивать изображения в БД по графическим запросам, представляющим собой какой-либо фрагмент искомого изображения.

Обработчик ЕЯ-запросов, табличный и анкетный интерфейсы осуществляют преобразование запросов и документов в некоторое внутреннее формальное представление, формируемое в терминах подсхемы пользователя. Интерпретатор моделей осуществляет перевод запросов и документов из этого внутреннего представления в представление в терминах схемы БД. Запросы и документы в терминах схемы БД однозначно интерпретируются СУБД.

Система управления базой данных обеспечивает доступ к информации в базе данных. ЛИНГВИСТ включает в себя СУБД ИРИС [Баскин и др., 1987]. БД имеет файловую структуру, файлы имеют иерархическую структуру и состоят из атрибутов, соответствующих реквизитам документов. Допускаются множественные атрибуты и объединение атрибутов в простые и периодические (повторяющиеся) группы. В каждом файле могут храниться до 64 000 записей. Каждая запись может иметь длину от 4 байт до 64 Кбайт. Максимальная длина отдельного атрибута составляет 32 Кбайта.

Компонент настройки на проблемную область осуществляет автоматизированное формирование, ввод, поиск, просмотр и корректировку проблемно-ориентированных знаний, используемых системой ЛИНГВИСТ для организации работы с документами пользователя. Компонент работает в рамках типовых диалогов в основном с анкетным интерфейсом.

При обработке текстовых реквизитов документов анкетный и табличный интерфейсы осуществляют индексирование текстов и обнаружение в текстах

Характеристики режимов	Режимы индексирования			
	Пословное	Основами	Дескрип- торами	Смешан- ное
Малый объем информационного обеспечения (ИО)	ИО не нужно	+	—	+
Большая скорость индексирования	+	+	—	+
Компактный поисковый образ	—	—	+	+
Выделение словосочетаний	—	—	+	+
Учет синонимии	—	—	+	+
Высокая полнота и точность поиска	—	—	+	+

Рис. 3.10. Сравнение режимов индексирования

орфографических ошибок. В ходе индексирования текстов проводится их полный морфологический анализ, выделяются информативные слова и словосочетания и формируются из них поисковые образы текстовых содержательных реквизитов документов и запросов. Это позволяет выполнять поиск документов по пословному вхождению поискового образа запроса в поисковый образ документа. Индексирование текстов производится в четырех режимах: 1) пословный режим — поисковый образ формируется как конъюнкция слов входного текста в том виде, как они встречаются в тексте (за исключением словформ, указанных в стоп-словаре системы); 2) режим неконтролируемой лексики — поисковый образ формируется как конъюнкция основ слов входного текста, выделенных в результате морфологического анализа (образуются все возможные основы, в том числе получаемые из исходной чередованием букв в основе, например «замысел», «замысла», «имя», «имени»); 3) режим контролируемой лексики — поисковый образ формируется как конъюнкция дескрипторов, приписанных в словаре основ словам и словосочетаниям входного текста; 4) смешанный режим — сочетание возможностей индексирования в режимах контролируемой и неконтролируемой лексики. Сравнительные характеристики различных режимов индексирования приведены на рис 3.10.

Орфографические ошибки обнаруживаются двумя способами: по буквенным сочетаниям словформ (полиграммный способ) и по словарю основ (словарный способ). При контроле по буквенным сочетаниям достигается меньшая вероятность обнаружения ошибки по сравнению с контролем по словарю основ. Однако данный способ обеспечивает высокое быстродействие, так как массивы разрешенных буквенных сочетаний невелики по объему и могут храниться во время работы целиком в оперативной памяти. Кроме того, контроль по буквенным сочетаниям пригоден для обнаружения ошибок в текстах, содержащих заранее неизвестные слова. Из-за большого объема словарь основ не может быть целиком загружен в оперативную память. Поэтому словарный способ контроля требует существенно больше времени, чем контроль по буквенным сочетаниям. Пользователь по своему усмотрению может применять тот и (или) другой способ. Выбор способа обнаружения орфографических ошибок и запуск его на исполнение осуществляются соответствующими функциональными клавишами. Один и тот же текстовый реквизит документа может быть последовательно обработан обоими способами.

Контроль по буквенным сочетаниям словформ осуществляется по буквосочетаниям длиной три (триадам). Поэтому в качестве массива разрешенных буквенных сочетаний используется словарь триад, хранящий все возможные (до-

пустимые и запрещенные) буквосочетания — отдельно триады начал слов, середины слов и концов слов. Для обнаружения ошибок каждое слово входного текста разбивается на триады. Например, для слова «завод» будут выделены триады «зав», «аво», «вод». Затем каждая триада проверяется на допустимость по словарю триад. Если в словаре триад данная триада помечена как запрещенная, то считается, что в слове допущена ошибка. Пользователь может пополнить словарь триад, пометив некоторую триаду как правильную.

Контроль по словарю основ в отличие от контроля по буквосочетаниям позволяет обнаруживать орфографические ошибки только в основах слов. Это не является существенным недостатком в силу следующих причин. Во-первых, вероятность ошибки в основах слов значительно больше, чем в окончании слов, так как основа слова обычно превосходит окончания по длине. Во-вторых, словарный способ обладает дополнительными возможностями: в случае одиночных орфографических ошибок, которые встречаются наиболее часто, система не только их обнаруживает, но и информирует пользователя о правильном написании слов. Одиночными называются ошибки типа пропуска одной буквы («заод» вместо «завод»), замены одной буквы другой («зовод» вместо «завод»), вставки одной лишней буквы («заввод» вместо «завод») и перестановки двух стоящих рядом букв («заовд» вместо «завод»).

3.2.2. Представление знаний. Основными знаниями, которые поддерживает и использует в работе система ЛИНГВИСТ, являются фактические предметные данные и знания, необходимые для обеспечения работы пользователя с предметными данными (системное информационное обеспечение).

Фактические предметные данные — это данные из документов пользователя. Именно они составляют область основных информационных интересов пользователя. Предметные данные хранятся в базе данных. На рис. 3.11 приведен пример файла БД. В этом файле содержатся фактографические данные о перевозке и добыче природных ископаемых (атрибут A1) по годам (A4). В периодические группы, показанные на рисунке двойными линиями, вынесены данные об объемах перевозок (A3) по видам транспорта (A2) и данные о добыче природных ископаемых (A6) по кварталам (A5).

Системное информационное обеспечение включает: сценарии диалогов; схему БД; подсхемы пользователя; правила интерпретации подсхемы пользователя на схему БД; знания о естественном языке пользователя; библиотеку описаний анкет; библиотеку графических примитивов и объектов. Сценарии диалогов хранятся в отдельных текстовых файлах в виде декларативных описаний (см. п. 3.2.3).

Файл1	A1	A2	A3	A4	A5	A6
	б_уголь	ж_тр	250	1987	1 кв	300
		реч_тр	80		2 кв	400
		мор_тр	100		3 кв	350
					4 кв	320
	нефть	ж_тр	180	1986	...	...
		.	.		.	.
		.	.		.	.
		.	.		.	.

Рис. 3.11. Пример файла БД

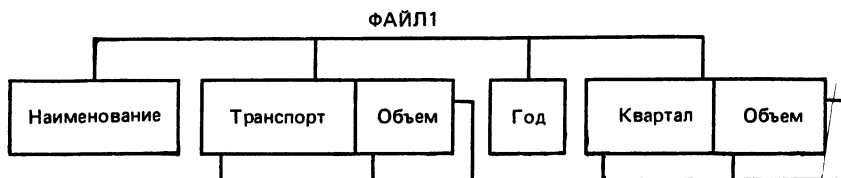


Рис. 3.12. Фрагмент модели данных системы

Схема БД является описанием логической структуры БД (рис. 3.12) и представляет собой совокупность отношений, соответствующих действительным файлам БД. Конечному пользователю работать непосредственно в терминах схемы БД не всегда удобно, так как, во-первых, его чаще всего интересует только часть (вырезка) информации из действительных отношений в БД, во-вторых, конкретного пользователя информация может интересовать не в том разрезе и виде, которые заложены в действительных отношениях БД. Поэтому ЛИНГВИСТ предоставляет пользователю возможность строить собственное представление (собственный взгляд) на данные [Ульман, 1983], хранимые в БД системы, т. е. подсхемы пользователя, и соотносить эти подсхемы со схемой БД с помощью правил интерпретации.

Подсхема пользователя — это описание в виде совокупности виртуальных (так как их физически может не быть в БД) и действительных отношений взгляда пользователя на структуру проблемной области (рис. 3.13). Для простоты использования предполагается, что виртуальные отношения имеют линейную структуру и взаимосвязаны через одинаковые значения атрибутов внутри отношений.

Схема БД и подсхема пользователя выражены в одинаковых представлениях. В них указываются: имя отношения; естественно-языковая интерпретация отношения (для объяснения смысла отношения); ключевые слова отношения (для поиска нужного отношения); автор и дата создания отношения; признак виртуальности; атрибуты отношения. Для каждого атрибута отношения описываются имя атрибута, естественно-языковая интерпретация атрибута, диапазон значений, признак поисковости и другая необходимая для функционирования системы информация. Так как схема БД и подсхема пользователя имеют одинаковые представления, то пользователь не чувствует разницы между виртуальными и действительными отношениями в БД.

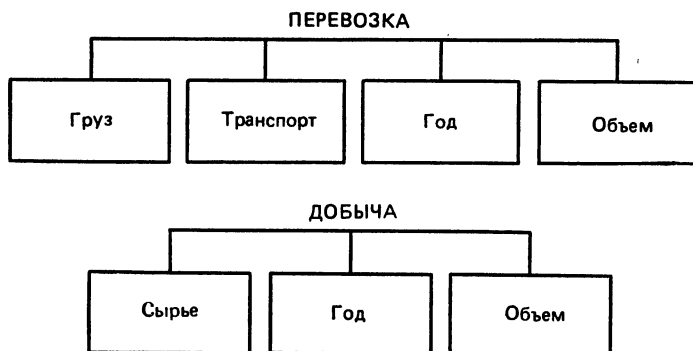


Рис. 3.13. Фрагмент модели данных пользователя

Основы	Слово-форма	Одушевленность	Часть речи	МТ	Дескриптор	Значение	Отношение	Атрибут	Падеж	Усеченные основы
01 завод	завод	н	с	01	заводы					авод заод звод завод заво
02 бур уголь	бурый уголь	— н	п с	31 12		б_уголь бурый	ПЕРЕ- ВОЗКА ДОБЫЧА	Груз Сырье	Род. Вин. Род. Вин.	
01 скольк	сколько	—	л	99		? ?	ПЕРЕ- ВОЗКА ДОБЫЧА	Объем год Объем год		

Рис. 3.14. Пример заполнения словаря основ

Правила интерпретации подсхемы пользователя на схему БД показывают, как выражаются виртуальные отношения через действительные отношения в БД. Правила имеют вид продукций:

виртуальное отношение ← действительные отношения;
условия; действия.

Например, виртуальные отношения ДОБЫЧА (см. рис. 3.13) связано с действительным отношением ФАЙЛ1 в БД (см. рис. 3.12) правилом:

ДОБЫЧА (сырье=& 11, год=& 12, объем=& 13) ←
ФАЙЛ1 (наименование=& 21, год=& 22, квартал=& 23, объем=& 24);
& 11=& 21, & 12=& 22;
& 13 присвоить СУММА & 24, для & 23=1 кв, 2 кв, 3 кв, 4 кв.

В приведенном правиле & NN обозначают имена атрибутов и используются для компактной записи условий и действий правил. В отношении ДОБЫЧА пользователя интересует годовой объем добычи сырья (& 13). Он может быть получен суммированием поквартальных объемов добычи (& 24) данного сырья (& 11=& 21) в данном году (& 12=& 22) в действительном отношении ФАЙЛ1. Таким образом, пользователю дается возможность работать с интересующим его отношением ДОБЫЧА, а система автоматически будет раскрывать это отношение через действительное отношение в БД.

Знания о естественном языке используются различными компонентами системы ЛИНГВИСТ для перевода запросов во внутреннее представление, индексирования текстов и обнаружения в текстах орфографических ошибок. Знания о естественном языке хранятся в словарях основ, триад и стоп-словаре.

Словарь основ имеет структуру, приведенную на рис. 3.14. Каждая запись словаря включает морфологическую и семантическую зоны и зону усеченных основ. Морфологическая зона служит для уточнения морфологической информации, описанной каждой основой, и выбора правильной основы из совокупности основ, построенных после отсекаания возможных окончаний. Семантическая зона словаря основ содержит значения из БД, соответствующие словам и словосочетаниям входного текста, и ссылку на отношения и атрибуты БД, значениями которых являются встречающиеся в тексте слова и словосочетания. Усеченные основы используются при словарном обнаружении ошибок. Каждая усеченная основа получается из основы слова путем удаления одной из букв. Например, для основы «завод» усеченные основы: «авод», «звод», «заод», «завд», «заво». Словарь триад используется для обнаружения орфографических ошибок. Он хранит в закодированном сжатом виде все возможные триады начал, середин и концов слов. Объем словаря триад 12 300 байт (каждая триада кодируется одним битом). Процесс формирования словаря основ и словаря триад автомати-

зирован. В стоп-словаре хранится список неинформативных слов, которые не обрабатываются при индексировании текстов.

В библиотеке описаний анкет хранятся описания анкет, используемые для ввода, поиска и просмотра документов пользователя. Описание анкеты производится на специальном языке, с помощью которого последовательно описываются все строки экранного представления анкеты. Каждая строка может состоять из фиксированного текста и полей для запоминания значений реквизитов документа. Поля могут описываться как невидимые, тогда они не отображаются на экране дисплея. Невидимые поля создают, например, для хранения и записи в БД производных значений, получаемых из значений других реквизитов документа. Система ЛИНГВИСТ осуществляет контроль заполнения полей анкеты. Поля могут быть описаны как содержащие только цифры, даты, русские тексты и любые тексты. Например, если поле описано как дата, то оно может содержать только цифры и точки в формате ЧЧ. ММ. ГГ, причем $0 < \text{ЧЧ} < 32$ и $0 < \text{ММ} < 13$. При попытке ввести символы, не соответствующие данному полю, раздается звуковой сигнал и высвечивается сообщение об ошибке. Фон анкеты, цвета символов и полей курсора могут быть заданы по усмотрению пользователя.

Библиотека графических примитивов и объектов содержит примитивы (квадрат, круг, эллипс, многоугольник и т. п.) и объекты, построенные пользователем из примитивов и других объектов. Графические объекты используются пользователем для построения более сложных изображений в документах пользователя.

3.2.3. Языковые и диалоговые возможности. ЛИНГВИСТ поддерживает взаимодействие с пользователем с помощью поликолонного экрана, что позволяет одновременно работать с разнотипными данными. Размер, цвет фона, цвет символов, местоположение окон на экране дисплея могут быть оперативно изменены и зафиксированы для дальнейшего использования. Окона могут накладываться одно на другое. В соответствии со структурой диалогов окна подразделяются на определенные типы и к каждому типу окон привязываются определенные процедуры обработки информации в окне. Тип окна отображается в его заголовке. При управлении работой системы основным типом окон является меню, к которому привязана процедура инициализации выбранного в меню действия. ЕЯ-запрос вводится в окно «ЕЯ-запрос», а для отображения результатов поиска по ЕЯ-запросу используется окно типа «таблица». В анкетном интерфейсе применяются окна типа «анкетный_запрос», «список», «анкета ввода», «анкета просмотра и корректировки», в табличном — типа «таблица», в графическом — типа «изображение».

Первая строка на экране отведена под сообщения системы, последняя — под краткую подсказку о назначении клавиш, инициирующих то или иное действие для текущего окна. Для любого текущего окна при нажатии клавиша F1 высвечивается окно «помощь». Это окно структурировано: в нем высвечивается описание решаемой пользователем задачи (вводится пользователем) и подробное описание клавиш, используемых при работе в текущем окне (может использоваться стандартное описание).

Ввод документа в БД осуществляется, как правило, в анкетной форме. Если документ содержит изображения, то из анкетного интерфейса вызывается графический интерфейс. Содержательные реквизиты документа вводятся, как обычные тексты на русском языке. Каждое предложение текста должно оканчиваться точкой. За каждым знаком препинания должен следовать пробел. Точки, ограничивающие инициалы и общепринятые сокращения, могут не отделяться пробелом от следующих за ними слов. Фактографические реквизиты документов представляют собой совокупности значений в виде слов и словосочетаний русского языка. Пример анкеты ввода приведен на рис. 3.9. Запрос на поиск может формулироваться на естественном языке в виде анкет, таблиц и изображений. Пример запроса в табличной форме приведен на рис. 3.8.

Диалоговые возможности системы ЛИНГВИСТ характеризуются тем, что общение с пользователем осуществляется в соответствии с заранее определенными сценариями. Сценарии диалогов позволяют организовать меню-ориентированное взаимодействие с системой. Каждый сценарий представляет собой совокуп-

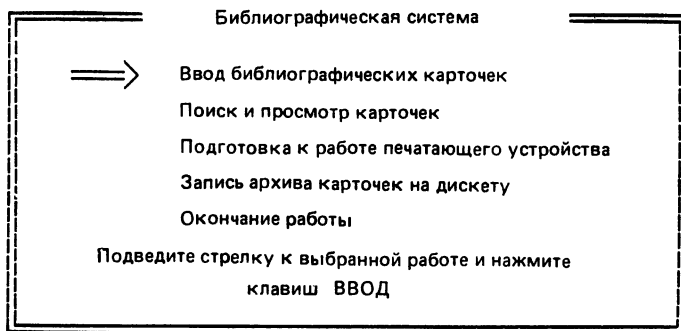


Рис. 3.15. Пример меню

ность описаний именованных меню и переходов из одного меню в другое. Вызов меню осуществляется по его имени, что позволяет создавать произвольный граф переходов из меню в меню. Через меню можно осуществить вызов окна любого типа с присоединенными к нему процедурами обработки данных и описать, какая информация должна быть помещена в окно (задать в описании меню имя описания для анкеты, имя таблицы и т. д.). Кроме того, через меню может быть вызвана любая программа пользователя. На рис. 3.15 приведен пример меню простейшей библиографической системы. Перемещение по меню осуществляется стрелками, а выбор работы — клавишем ВВОД.

Система предлагает типовые сценарии диалогов, которые пользователь при необходимости может изменить. Особенностью построения типовых диалогов является то, что система ведет пользователя к достижению цели наиболее традиционным в текущей ситуации способом, а пользователь в любой момент может навязать свой план достижения цели. Например, ЛИНГВИСТ всегда после создания файла в БД предлагает ввести информацию в созданный файл (курсор после создания файла устанавливается на строке в меню, обозначающей работу по вводу информации). Пользователь по своему усмотрению может избирать и другую работу из предложенных в текущем меню.

В каждом окне, инициализированном выбором соответствующей работы в меню, предусмотрены определенные действия. Выбор действия осуществляется функциональными клавишами. Рассмотрим типовые действия при работе с окнами ввода, поиска и просмотра документов в анкетном интерфейсе на примере библиографической системы. Заполнение библиографических карточек производится в окне «анкета ввода». В начальный момент в этом окне высвечивается пустой бланк документа. Пользователь заполняет его (см. рис. 3.9) и, нажав определенный клавиш, записывает документ в БД, после чего высвечивается новый пустой бланк. В ходе заполнения документа пользователь может перемещать строки документа в окне, курсор по полям, исправлять введенные в документ данные и т. д. При вводе длинных текстов в поля бланка автоматически вставляются новые строки.

Процесс поиска в архиве карточек позволяет искать библиографические карточки в БД по различным условиям, заданным пользователем, а также просматривать и корректировать найденные документы. В начальный момент пользователю высвечивается бланк запроса на поиск. Заполняя поля этого бланка, он формирует запрос на поиск документов. Пример бланка запроса приведен на рис. 3.16. Если пользователь в своем запросе заполнил несколько полей бланка, то это означает, что требуется найти те документы, у которых значения соответствующих реквизитов удовлетворяют значениям заполненных полей. В примере на рис. 3.16 по запросу необходимо найти литературу за 1986 год о локальных вычислительных сетях.

Запрос на поиск в архиве	
Тематика	_____
Автор	_____
Заглавие	_____
Издательство	_____ Год издания 1986 _____
Аннотация	О локальных вычислительных сетях _____

Рис. 3.16. Пример анкетного запроса

После выполнения заданного запроса пользователь может получить полное изображение каждого найденного документа, используя при этом тот же бланк, в котором вводился документ, или бланк, специально созданный для просмотра. Документ может быть отредактирован и вновь записан в БД. Пользователь может получить также компактный список, содержащий часть полей из найденных по запросу документов. Пример списка, построенного в результате поиска в библиографической системе, приведен на рис. 3.17. Перемещая курсор по строкам списка, пользователь может выбрать документ, нужный ему для работы, и получить его полное изображение.

3.2.4. Возможности настройки. Эти возможности определяются декларативно представленными знаниями. Настройке подлежат лексика языка общения, логическая структура (схема) БД, представление пользователя о хранимых данных, сценарии диалогов. Пользователь может выбрать приемлемый язык общения (естественный язык, анкеты, изображения, таблицы), создать персональный каталог запросов, применять персональную лексику.

Возможности настройки обеспечивают компонент настройки на проблемную область. С его помощью пользователь создает: 1) схему БД и файлы БД; 2) подсхемы пользователя (для табличного интерфейса и обработчика ЕЯ-запросов); 3) правила интерпретации подсхем пользователя на схему БД (для табличного интерфейса и обработчика ЕЯ-запросов); 4) сценарии диалогов (можно выбирать типовые сценарии); 5) описания анкет (для анкетного интерфейса); 6) словарь основ (морфологическая и семантическая зоны — для обработчика ЕЯ-запросов, а также для индексирования текстов, зона усеченных основ — для обнаружения орфографических ошибок словарным способом); 7) словарь триад для обнаружения орфографических ошибок триадным способом), 8) библиотеку графических примитивов и объектов (для графического интерфейса).

Все работы по настройке системы проводятся в рамках типовых диалогов с помощью анкетного интерфейса. От пользователя не требуется специальных знаний, в частности, в области лингвистики. Часть данных в словарях формируется автоматически (например, усеченные основы, триады), некоторые — с привлечением пользователя (например, для определения основы слова и морфологического типа система просит поставить слово в определенных падежах). Пользователь может работать с одним из видов интерфейсов, предоставляемых системой, и при настройке системы вводить только те знания, которые необходимы для работы этого интерфейса. Так, в простейшем случае для работы в рамках табличного интерфейса необходимо создавать схему БД и файлы БД, а для работы в рамках анкетного интерфейса — схему БД, файлы БД и описания анкет.

3.2.5. Реализация системы. Система ЛИНГВИСТ реализована и предназначена для работы на ПЭВМ типа ЕС-1841 (и последующих моделях), IBM PC/XT, AT и т. п. со стандартной операционной системой ДОС-16 (MS-DOS). Программное обеспечение создано на языке программирования Си и частично — на Ассемблере. Для работы системы необходимо 300 Кбайт оперативной памяти.

Список найденных библиографических карточек

Номер	Заглавие	Автор	Год изд.
19	Вопросы экономического и _____ организационного обеспечения функционирования сетей _____ вычислительных центров _____	Гафт Л. Ш. _____ Зарубин И. И. _____	1986
48	Перемены на рынке локальных сетей _____	Розенберг Р. _____	1986
67	Новое поколение локальных _____ сетей низкого быстродействия _____	Иванов А. А. _____	1986

Рис. 3.17. Пример списка документов

Система ЛИНГВИСТ использовалась для создания ряда информационных систем на ПЭВМ. В частности, на основе системы ЛИНГВИСТ автоматизировано ведение историй болезни стоматологических больных. История болезни одного больного представляет собой совокупность 15 взаимосвязанных документов различной формы (паспортные данные, результаты анализов, дневник лечения и т. д.) с 1500 реквизитами, многие из которых носят описательный характер (диагноз, рекомендации и т. п.). Лингвистический словарь для индексирования содержательных реквизитов в этой задаче содержит около 5000 словарных статей.

Документация на систему ЛИНГВИСТ выполнена в соответствии с требованиями ЕСПД. В состав эксплуатационной документации входят описание применения, руководство пользователя и руководство программиста. Система поставляется на пяти гибких магнитных дисках (360 Кбайт каждый). Первая версия системы сдана в СФАП Алгоритм в 1988 г.

3.3. Промышленная система ДИСПУТ

Л. И. Микулич

3.3.1. Общее описание. Система ДИСПУТ [Брябрин и др., 1983; Mikulich, 1982] предназначена для обеспечения интеллектуального интерфейса непрограммирующего профессионала с базами данных.

Система дает возможность пользователю: 1) задавать вопросы на естественном языке, ограниченном прагматикой проблемной области; 2) формировать и вести персональный каталог запросов, определенных пользователем как стандартные или типовые; 3) вести информационные массивы системы; 4) персонифицировать словарь лексики на уровне синонимии; 5) работать с библиотекой моделей.

Название системы возникло из заглавных букв полного названия первой версии системы и характеризовало решаемую прикладную задачу: Диалоговая Информационно-справочная Система для Планирования и Управления на Транспорте. В дальнейшем система развивалась автономно и использовалась для решения других задач, но первоначальное название сохранила.

Область применения системы ограничена проблемной средой с достаточно простой структурой отношений между объектами этой среды, в которой профессиональная лексика содержит до 1000 неконстантных лексических единиц. Под неконстантными лексическими единицами понимаются слова профессионального языка пользователя, не являющиеся именами собственными в рамках прагматики данной проблемной области. Например, слова, входящие в понятие «груз» («кофе», «железная руда», «трубы», «ткани» и т. п.), относятся к константам,

так как они являются значениями атрибута «груз» Само же слово «груз» является-неконстантным.

Под достаточно простой структурой отношений между объектами подразумевается такая среда, сведения о которой могут быть приведены к множеству однородных пар «атрибут-значение». Эти пары образуют информационные массивы (или базу данных) системы.

На рис. 3.18 дана типовая схема систем типа ДИСПУТ версий до 1.5 включительно. Система представляет собой набор модулей, схема управления которыми организована по иерархическому принципу. Этот принцип соблюдается во всех версиях строго, хотя во многих из них для удобства пользователя имитируется перекрестное управление (сетевая структура). Головная управляющая программа (монитор) системы подчиняет себе мониторы второго уровня. Каждый из этих мониторов, в свою очередь, руководит работой модулей в процессе выполнения одного из видов работ.

Основная функция монитора системы состоит в организации последовательности вызовов мониторов работ и контроля логики переходов из одного монитора второго уровня к другому. В функции монитора системы входит также вызов модуля настройки системы на пользователя, вызов сервисных модулей и операция завершения работы системы.

Модуль настройки системы на пользователя служит для идентификации пользователя по его фамилии и паролю и открытия соответствующих рабочих и информационных массивов. При этом пользователь получает доступ к своему или общему каталогу типовых запросов, к интерпретатору (или интерпретаторам) запросов и соответствующим словарям лексики и информационным массивам. В зависимости от статуса пользователя ему могут быть доступны все или только некоторые виды работ в пределах нескольких или всех мониторов второго уровня.

Монитор работы с каталогом типовых запросов организует вызов и исполнение программ, обеспечивающих различные варианты работ с каталогом. Если пользователю при генерации был открыт персональный каталог запросов, то ему доступен весь список работ с каталогом.

Монитор каталога допускает следующие виды работ: 1) просмотр каталога; 2) выбор запроса по его номеру в каталоге; 3) добавление нового запроса в каталог; 4) исключение запроса из каталога; 5) вывод текста каталога на устройство широкой печати.

Перед началом работы с каталогом система определяет его параметры, которые выводятся на экран пользователю. Эти параметры включают число хранимых запросов и число свободных строк для записи новых запросов. Система позволяет хранить до 99 запросов в каждом каталоге, причем длина каждого запроса не должна превышать 240 символов. Эти ограничения были выбраны произвольно исходя из анализа запросов, реально возникающих при решении производственных задач.

В процессе просмотра каталога пользователь может выбрать нужный ему запрос и отправить его на исполнение. Отдельный вид работы «Выбор запроса» введен для того, чтобы пользователь мог извлекать запрос по его номеру без просмотра содержимого каталога. Если выбранный из каталога запрос содержит свободные параметры, то они должны быть означены (т. е. получить значения) перед отправкой его на исполнение. В этот момент запрос может быть также изменен произвольным образом.

При выполнении операции включения нового запроса его текст запрашивается у пользователя и в канонической форме заносится в каталог. Новому запросу присваивается очередной номер, и он помещается в конец списка запросов. Заносимый в каталог запрос может быть сразу же отправлен на выполнение.

Монитор работы с информационными массивами предоставляет пользователю следующие виды работ: 1) выбор массива; 2) просмотр массива; 3) коррекция записей массива; 4) добавление новой записи в массив; 5) удаление записи из массива; 6) пересылка записи из одного массива в другой.

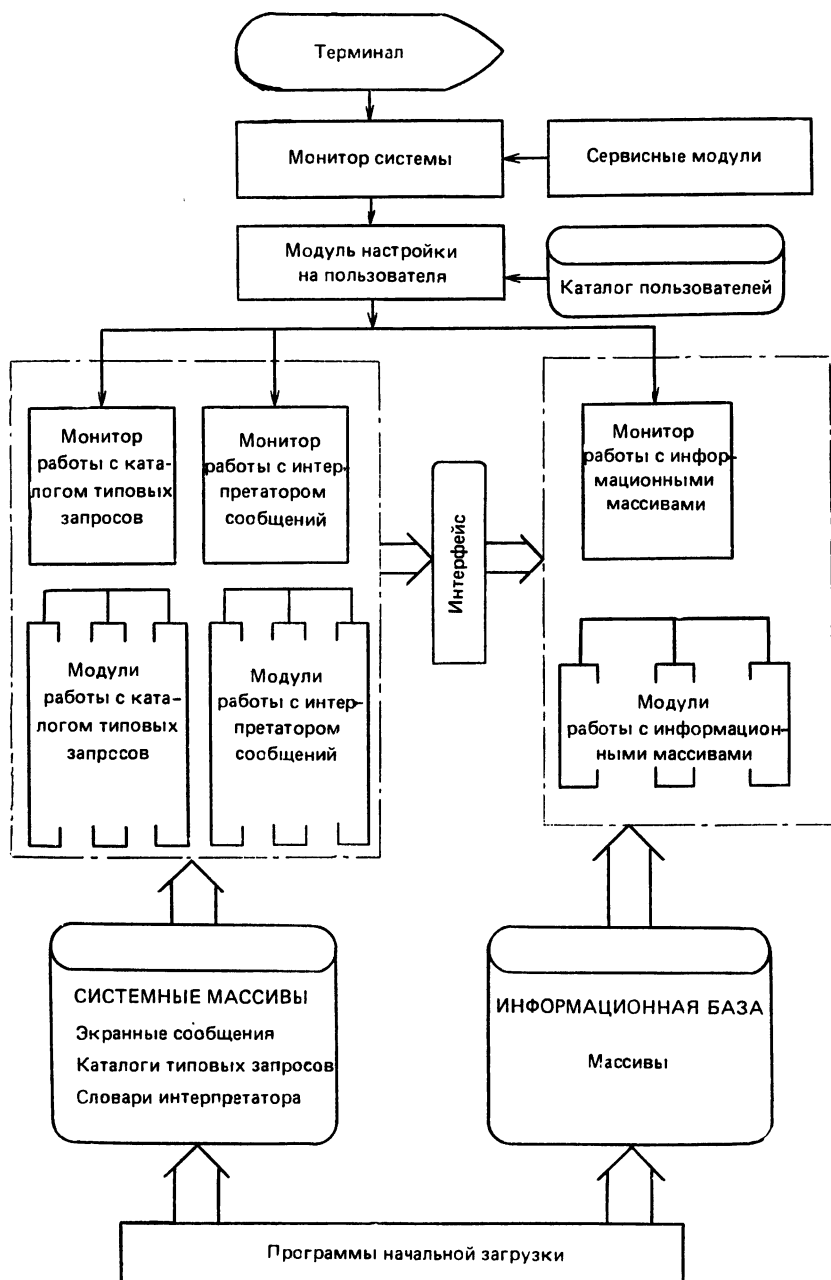


Рис. 3.18. Структура системы ДИСПУТ

Информация, относящаяся к проблемной области, может храниться в нескольких автономных массивах. Система позволяет осуществлять выбор конкретного массива для последующей работы с ним. На системном уровне это означает открытие соответствующего файла.

Каждый из остальных видов работ с массивами мало отличается от функций традиционных редакторов массивов. Существующие отличия состоят в том, что вводимое или изменяемое значение любого параметра записи проходит синтаксическую и семантическую проверку перед его записью в файл. Семантика проверок привязана к семантике проблемной области. Как правило, программы таких проверок существуют автономно и обслуживаются в рамках базы данных, с которой работает система ДИСПУТ.

Некоторое отличие от традиционных имеет функция пересылки записи из одного массива в другой. Семантика этой функции также оговорена заранее. Она предполагает извлечение из одного массива только тех параметров (реквизитов) вместе с их значениями, которые будут необходимы для записи в другой массив, а также добавление системой требуемых реквизитов и их означивание пользователем перед записью в другой массив.

Монитор работы с интерпретатором сообщений осуществляет: 1) считывание в основную память словарей системы, хранящихся на диске; 2) последовательный вызов модулей, выполняющих интерпретацию текста на естественном языке в терминах языка манипулирования данными; 3) запуск программ работы с информационными массивами; 4) трассировку всех операций, выполняемых системой в этом режиме, начиная от поступления вопроса и кончая получением ответа.

Интерпретатор запросов включает в себя два комплекса программ, называемых «процессорами»: лингвистический и прагматический. Первый используется для перевода естественно-языковых конструкций во внутреннее формальное представление, инвариантное смыслу запроса. Во второй переносятся знания о прагматике задачи, позволяющие эффективно использовать структуру базы данных и набор обрабатываемых процедур для получения ответа.

Модули лингвистического процессора выполняют следующие функции: 1) предморфологический анализ; 2) кодирование текста в соответствии со словарем лексики; 3) предсинтаксический анализ; 4) семантико-синтаксический анализ.

В процессе предморфологического анализа выделяются знаки препинания и специальные символы, поскольку они рассматриваются как автономные лексические единицы.

На этапе кодирования происходит пословное сопоставление вводимого запроса со словарем лексики, и каждой входной словоформе ставится в соответствие ее синтактико-семантический код.

В процессе предсинтаксического анализа происходит обращение к словарю сверток. С его помощью в запросе распознаются устойчивые лексические конструкции, которые имеют фиксированную семантическую интерпретацию в данной проблемной области. В результате предсинтаксического анализа для дальнейшей работы сохраняются только те коды словоформ и сочетаний, которые необходимы для однозначного понимания входного предложения.

Основной анализ запроса осуществляется в семантико-синтаксическом анализаторе, в котором реализована грамматика, обслуживающая конкретную проблемную область, — прагматическая грамматика. Результатом работы этого модуля является формальное внутреннее представление запроса.

Модули прагматического процессора реализуют следующие функции: 1) предпрагматический анализ; 2) прагматический анализ; 3) формирование задания на поиск.

При работе модуля семантико-синтаксического анализа (в лингвистическом процессоре) осуществляется вызов ряда модулей прагматического процессора. Этим обеспечивается, практически во всех случаях, одновариантный анализ входного предложения за счет использования прагматической информации о проблемной области.

Модули работы с информационными массивами предназначены для исполнения задания пользователя, содержащегося в его запросе. В их функции входят: 1) планирование работ по поиску ответа; 2) поиск в информационных массивах; 3) формирование массива для сортировки; 4) выполнение арифметических и логических операций над элементами поиска; 5) сортировочные операции.

При работе со стандартной СУБД большинство из перечисленных выше функций может быть заимствовано из этой СУБД. При самостоятельном проектировании набора информационных файлов используются функциональные модули работы с ними, содержащиеся в комплексе системы ДИСПУТ.

Генерация и оформление ответа выполняются несколькими модулями. Они осуществляют формирование заголовков таблиц и их вывод вместе с соответствующим информационным наполнением на экран и широкую печать. При необходимости возможна выдача экранной информации на принтер.

3.3.2. Способы представления знаний в системе. О способах представления знаний в системе будет говориться применительно к интерпретатору, поскольку остальную часть системы вряд ли целесообразно рассматривать в рамках парадигмы искусственного интеллекта. Интерпретатор содержит следующие виды знаний: 1) лингвистические; 2) прагматические; 3) метазнания о самой системе.

Перечисленные типы знаний, взятые в отдельности, не имеют четкой локализации в системе и однозначного способа представления. Иными словами, знания в основном компилированы и введены в тексты программ, реализующих интерпретатор.

Часть знаний, связанных со словарями, в разных версиях системы имеет как компилированную форму, так и декларативную, вынесенную во внешние файлы. Такой способ реализации знаний в системе имеет как положительные, так и отрицательные стороны. Главным положительным моментом распределенных компилированных знаний является высокая эффективность получаемой системы. Время интерпретации запроса в таких версиях составляет доли секунды для любой допустимой его конструкции. Отрицательных моментов два: возрастающие емкости требуемой основной памяти для размещения системы и потеря гибкости при перенастройке. Поэтому в последних версиях существует тенденция к увеличению доли декларативных знаний в системе за счет снижения скорости интерпретации, но в пользу повышения ее адаптационных свойств и большей компактности.

В лингвистическом процессоре реализована прагматическая грамматика. Эта грамматика описывает синтаксические и ситуативные конструкции языка пользователя, работающего с системой в конкретной предметной области. Разные проблемные области и решаемые в них задачи описываются разными прагматическими грамматиками. Сужение проблемной области и класса решаемых в ней задач приводит, как правило, к упрощению грамматики. При этом язык, которым пользуется человек при работе с системой, всегда есть естественный профессиональный язык пользователя, и прагматическая грамматика описывает этот язык.

Прагматическая грамматика отображена в: словаре лексики; словаре устойчивых словосочетаний; концептуальных фреймах; схеме анализа предложения.

Каждой словоформе входного текста словарь лексики ставит в соответствие ее описание, которое включает в себя: 1) минимальное допустимое сокращение словоформы, оставляющее ее в классе попарно различных слов для данной проблемной области; 2) синтаксическую категорию (опорное слово, переходный глагол, непереходный глагол, атрибут, обстоятельство, предлог и т. д.), которая может не совпадать с традиционными лингвистическими категориями естественного языка, но имеет смысл в рамках заданной прагматики; 3) признак участия в словосочетаниях.

Вся эта информация представлена в виде числового кода, в соответствии с которым происходит дальнейшая работа системы. Решение о таком кодировании словарной информации первоначально было принято в связи с выбранным языком реализации системы, но оказалось удобным и экономичным при реализации и на других языках программирования.

Словарь устойчивых словосочетаний содержит конструкции, каждая из которых имеет фиксированную семантику в данной проблемной области и может рассматриваться как одна смысловая единица.

Концептуальные фреймы задают темы или типы запросов, которые система будет способна анализировать нужным образом. Имена слотов этих фреймов соответствуют глубинным падежам, хотя и не следуют грамматике Филлмора.

В процессе функционирования системы выбирается один из концептуальных фреймов, слоты которого означаются в соответствии с заданной схемой анализа предложения. Эта схема определяет порядок выбора фреймов и работы с присоединенными к слотам процедурами, некоторые из которых сами имеют фреймовую структуру.

Результатом работы лингвистического процессора является фрейм-экземпляр, все или большинство слотов которого означены.

Знания прагматического процессора используются для преобразования фрейма-экземпляра, полученного из лингвистического процессора, в набор параметров планировщика, строящего траекторию поиска ответа. Прагматический процессор содержит знания о проблемной области в виде набора имен объектов и отношений между ними; о базе данных, включая знания о структуре массивов и их реквизитном наполнении; об операциях над массивами и выборками из них, допустимыми в системе.

Эти знания хранятся как в виде прагматических словарей (таблиц), так и в виде компилированного набора продукций.

Работа прагматического процессора может рассматриваться как совокупность действий группы «специалистов» («экспертов»), которые отыскивают соответствие между слотами входного фрейма и описаниями проблемной области и/или базы данных, производя при необходимости простой логический вывод. Они определяют вид функции, которая задает способ обработки информации, извлеченной из базы данных. Они работают с исключениями, где перечисляются все специальные случаи, которые не позволили означить некоторые слоты фрейма в лингвистическом процессоре.

Наконец, «эксперты» могут возратить анализ назад в лингвистический процессор, переназначив ему концептуальный фрейм. Такие случаи могут возникнуть, например, когда имена некоторых массивов совпадают с именами реквизитов в других массивах.

Метазнания системы включают знания: о синтаксисе и грамматике, т. е. позволяют уточнять или выбирать типы вопросов, содержат сведения о грамматических конструкциях, допустимых и недопустимых в системе, об ограничениях и условиях, принятых в системе; о прагматике, т. е. о структуре проблемной области, понятиях и отношениях, о структуре базы данных, о массивах и их реквизитном наполнении, о реализованных в системе функциях над наборами данных и выборками; о дискурсе, т. е. о моментах смены инициативы в диалоге и необходимых условиях для этого; о партнере диалога, т. е. о пользователе, его статусе, знании системы, умении с ней работать, его психофизиологических характеристиках.

Метазнания реализованы в виде наборов продукций, большей частью работающих с табличной информацией, но использующих также статистические данные сеансов для изменения некоторых знаний (например, о пользователе).

3.3.3. Языковые возможности. Принцип разработки системы, который исходит из примата прагматики и учета лингвистических особенностей каждого из потенциальных пользователей системы, определяет и ее языковые возможности. Словарь лексики системы создается на основе анализа профессионального языка пользователей и строится как семантическое объединение их персональных жаргонов. На уровне входного языка каждый из них получает для пользования свой собственный и удобный для него диалект. Он извлекается из памяти в момент идентификации пользователя системой. В процессе работы с системой словарь лексики пользователя может расширяться на уровне синонимии.

С точки зрения традиционной грамматики допустимыми являются следующие конструкции: простые предложения; сложноподчиненные предложения или предложения, содержащие причастные обороты.

Как правило, в предложениях не допустимы разрывные составляющие. Следует подчеркнуть, что упомянутые конструкции не были выбраны произвольно, а явились результатом анализа запросов реальных пользователей при решении ими повседневных профессиональных задач.

В некоторых системах на входной язык налагаются достаточно простые ограничения и допускаются определенные условности. Примером ограничения и одновременно условности является необходимость обращения кавычками имен собственных или слов/словосочетаний, приравниваемых в данной проблемной области к константным терминам. Например, это могут быть нечисловые значения, которые принимают некоторые реквизиты базы данных. К условностям можно отнести и допустимость расстановки круглых скобок для обозначения вложенных конструкций вместо употребления словосочетаний типа «в том числе ...» или «по ...».

Конструкция словаря лексики позволяет формулировать запросы, не используя полные написания слов. Возможно их сокращение до определенных пределов (для каждого слова — свое). Возможность написания сокращенных слов влечет за собой толерантность к ошибкам и опiskaм, приходящимся на «незначущую» часть слова.

Запросы формулируются пользователем в привычном ему стиле и на его собственном жаргоне и размещаются в любом месте экрана, кроме защищенных полей.

3.3.4. Диалоговые возможности. В начале работы системы диалог проходит по традиционной схеме с инициативой у машины. В качестве языка диалога используется язык меню и краткий набор команд, единых для всей системы. При работе с интерпретатором инициатива передается человеку и вновь может быть перехвачена системой в критических точках, т. е. в тех ситуациях, когда пользователь начинает совершать неадекватные действия или когда анализ запроса заходит в тупик.

В системе предусмотрена возможность задавать уровень ее «диалоговости» (или «болтливости»). На начальных стадиях работы с системой пользователю, который задает много нерегламентированных запросов, бывает целесообразно устраивать прерывания для выдачи промежуточных результатов. Эти результаты могут содержать информацию о том, как понял системой запрос, какие массивы назначены для поиска и т. п. При этом система способна воспринимать и обрабатывать ответы пользователя и менять свое поведение в зависимости от их содержания. По мере привыкания к системе ее чрезмерная разговорчивость начинает тяготить пользователя, и он может отказаться от частых собеседований с системой.

Регулировать уровень диалоговости можно как с помощью каталога пользователей, где фиксируются все промежуточные интерфейсы для конкретного пользователя, так и путем явных указаний в процессе диалога.

Как уже отмечалось, рассматриваемые версии системы не работают со связным текстом. Поэтому полноценного дискурса в системе не предусмотрено. Диалог можно вести относительно одного отдельно взятого запроса, например, о правильности или неправильности его задания, о возможных вариантах поиска ответа, т. е. с привлечением той информации, которая составляет метазнание системы.

Система способна обрабатывать некоторые анафорические конструкции, относящиеся к одному запросу. Однако эти возможности реализованы в ней не как правило, а как исключение.

Работая с интерпретатором, пользователь может перехватывать инициативу и выходить на работу в других режимах: работу с каталогом, с массивами, моделями и т. д.

3.3.5. Возможности настройки и адаптации. Одной из главных задач при разработке рассматриваемой системы была ее эффективность. Отсюда стремле-

ние удалить из системы все, что может эту эффективность уменьшить. Как уже отмечалось, большинство знаний системы компилировано и не доступно для непосредственной модификации из самой системы. Поэтому настройка системы на новую проблемную область сопряжена с перекомпиляцией значительной части системы, выполняемой с помощью специальных средств, не входящих в саму систему. Эта операция выполняется только разработчиками и не может выполняться заказчиками и пользователями.

Адаптация системы к пользователям предусмотрена в системе и может осуществляться штатом программистов, сопровождающих систему на вычислительном центре, где она установлена. Адаптация включает формирование персонального словаря, уровней доступа и построение психологического портрета пользователя, определяющего форму и структуру диалогового режима.

3.3.6. Сведения о реализации системы. Основное множество задач, для решения которых использовалась система ДИСПУТ, относится к задачам управления перевозочным процессом на морском транспорте. Реально функционирующие промышленные системы решают именно эти задачи. Однако в экспериментальном варианте, не доведенном до промышленного внедрения, система испытывалась в задачах интерпретации данных океанографических наблюдений, задачах учета и распределения материалов на предприятии, в кадровых системах.

В реальных ситуациях, где использовалась система, словари лексикис содержали до 500—700 словарных единиц. Кроме того, словари словосочетаний включали до 300—350 сверток. Разные версии системы имеют две основные реализации: одна предназначена для работы на ЭВМ типа ЕС (наибольшее количество версий), другая реализована на ЭВМ типа Реалите (одна версия).

Языками реализации для машин типа ЕС являются Фортран-IV и Ассемблер, а для машин типа Реалите — язык Реалите-Бейсик. На Ассемблере написаны планировщик, интерфейс с СУБД, модули, реализующие функции работы с выборками из массивов, и генератор ответов. Все остальные модули реализованы на Фортране.

Все программное обеспечение на машинах типа ЕС занимает не более 600 Кбайт памяти. Во время исполнения задания в основной памяти хранится от 120 до 320 Ккодов. Требуемая емкость определяется режимами редактирования модулей (с перекрытием, без перекрытия), необходимостью размещения в памяти интерфейсного модуля для работы с базой данных, который может занимать значительный объем (например, при работе СУБД ИНЭС-2), и объемом сортировочных буферов. Системные файлы, хранимые на дисках, требуют не более трех треков на НМД типа ЕС-5061.

Время реакции системы, т. е. время от нажатия на клавиш ввода до появления сообщения на экране, составляет от 0,5 до 1,7 с в режиме интерпретации запроса. Время поиска ответа существенно зависит от используемой СУБД и объемов информационных массивов. Практически это время колеблется от 10 с до 2—3 мин для запросов, не требующих многоуровневой сортировки, при базе, содержащей около 50 000 записей по 25—30 реквизитов каждая на ЭВМ типа ЕС-1055 или ЕС-1045. Использование стандартных процедур сортировки, входящих в операционную систему, добавляет по 20—30 с на каждую сортировку.

Требуемая операционная система — ОС ЕС версий 4.1 и последующих.

Система работает с дисплейными станциями типа ЕС 7906 или ЕС-7920. Возможна работа с любыми аналоговыми станциями.

Система поставляется на дистрибутивной ленте, содержащей следующие файлы: библиотеку загрузочных модулей; библиотеку заданий; библиотеку исходных модулей, необходимых при генерации системы; системные файлы; макет каталога; контрольный пример; описание применения; руководство для программиста; руководство для оператора; руководство для пользователя.

Теоретические основы систем речевого общения

Б. М. Лобанов

4.1. Основные положения

В системах искусственного интеллекта с элементами естественно-языкового (ЕЯ) общения обычно предполагается, что в качестве средства общения используется письменная речь. По понятным причинам это не всегда удобно, а во многих случаях и неэффективно. Использование устной речи как средства общения позволяет почти на порядок повысить скорость ввода информации, разгрузить зрение и освободить руки, осуществить речевое общение на значительном расстоянии и по телефону. Если не затрагивать общих проблем ЕЯ-общения, которые связаны с пониманием речи, с созданием преобразователей «смысл — текст» и «текст — смысл», то узко специальными проблемами, стоящими перед разработчиками систем речевого общения (СРО), становятся проблемы создания преобразователей «текст — речевой сигнал» и «речевой сигнал — текст». Первая из них называется проблемой *синтеза речи*, вторая — *анализа и распознавания речи*.

В системах ЕЯ-общения под текстом обычно понимают *орфографический*, или буквенный (как пишется), текст, в СРО — *фонемный* (как слышится). В создании преобразователей орфографического текста в фонемный и наоборот не существует особых проблем, хотя сложность таких преобразователей для разных языков будет различной (ср. русский и английский). Поэтому применительно к СРО можно ограничиться проблемами разработки преобразователей «цепочка фонем — речевой сигнал» и «речевой сигнал — цепочка фонем». В речи фонема выступает в двух аспектах. С одной стороны, это элементарная смысловозначительная единица письменной речи, с другой — абстрактное обозначение конкретного звука устной речи. Отметим сразу, что не существует счетного множества, а тем более одного-единственного звука речи, соотносимого с фонемой¹. Если учесть, что на речевой звук налагается множество экстралингвистических факторов: индивидуальные особенности речи и голоса, эмоциональное и физиологическое состояния говорящего, электроакустические характеристики среды и тракта передачи, а также шумы, помехи и искажения, то сложность проблемы, стоящей перед создателями СРО, пожалуй, трудно переоценить.

Современные исследования в области СРО начаты в индустриально развитых странах в начале 60-х годов. Первые промышленные СРО появились в конце 70-х годов. К настоящему времени созданы разнообразные СРО для разных сфер применения. Это связано с осознанием потенциальными потребителями преимуществ СРО:

удобство, простота и естественность процедуры общения, требующей минимума специальной подготовки;

возможность использования для связи с ЭВМ обычных телефонных аппаратов и существующей телефонной сети;

устранение ручных манипуляций с одновременным увеличением скорости ввода информации (в 3—5 раз по сравнению с клавиатурным вводом) и разгрузка зрения при получении информации.

Для иллюстрации того, как могут быть использованы или уже используются каждое из трех указанных преимуществ, приведем некоторые конкретные примеры применения СРО.

Первое и второе преимущества с наибольшим эффектом проявляется в автоматизированных системах управления (АСУ) предприятия, организации или отрасли. Однако пока АСУ выполняет в основном информационные функции.

¹ Жинкин Н. И. Механизмы речи, — М.: АПНР РСФСР, 1958, — 370 с.

Одной из причин такого положения является отсутствие массового оперативного контакта человека и ЭВМ. В результате чрезвычайно затрудняется выполнение основной функции системы — управления. С внедрением в АСУ речевой технологии общения ЭВМ станет по-настоящему активным звеном управления. Круг пользователей системы в силу простоты и естественности общения с ЭВМ неизмеримо расширится. При этом человеко-машинное взаимодействие будет осуществляться с помощью местной или городской телефонной сети на значительном расстоянии. В любое время большая группа пользователей сможет одновременно обращаться к ЭВМ за получением разного рода нормативно-справочной информации. Имея необходимую базу знаний — модель каждого конкретного производства, система сможет по запросу пользователя выдавать необходимую консультацию по телефону в случае возникновения трудностей в той или иной сфере производства. Кроме того, система сможет выполнять функции диспетчера, обращаясь по телефону в необходимое время к конкретным работникам за получением данных о ходе производства, предупреждая их о наступлении критических ситуаций, напоминая о сроках поставки изделий или выполнения разного рода обязательств, а также осуществлять учет и контроль. Несомненно, во всех этих случаях эффективная работа СРО возможна лишь при условии, что она является составной частью систем искусственного интеллекта.

Третье свойство — разгрузка зрения и рук — важно для создания нового поколения систем оперативного человеко-машинного управления сложными объектами. К ним относятся прежде всего системы управления движением, энергетическими установками и другие АСУ ТП, САПР и АРМ. В эпоху, когда интенсивность движения и сложность объектов становятся критическими, любая новая возможность повысить оперативность управления и снизить риск аварии должна быть тщательно изучена и использована. Немедленное получение от компьютера информации в речевой форме и немедленный ввод речевой команды, безусловно, будут важным дополнением к традиционным системам световой индикации и клавишного управления. При этом получение и передачу информации можно будет осуществлять не только с рабочего места оператора, но и на некотором удалении или даже во время движения.

Говорящие компьютеры можно будет с успехом применять и в непромышленной сфере, например при обучении родному языку или иностранным языкам. В дальнейшем по мере совершенствования техники машинного перевода появится уникальная возможность синхронного устного перевода с одного языка на другой.

Можно также отметить применение речевой технологии в медицинских целях. Появятся говорящие книги и компьютеры для слепых, управляемые голосом, инвалидные коляски, приборы для генерации и восприятия речи глухонемых и др.

Основой для разработки современных СРО является лингвоакустическая и информационная теории речеобразования и восприятия речи. Лингвистическая теория рассматривает фонетические и просодические характеристики речи, акустическая — акустические характеристики (признаки) фонем и просодем, информационная — структуру речевого сигнала. Эффективность СРО тем выше, чем полнее реализованы в ней принципы функционирования естественной речевой системы человека. При этом не обязательно, чтобы искусственные СРО копировали структурные особенности работы естественной системы. Важно, чтобы СРО как можно ближе была в функциональном отношении к естественной.

4.2. Фонетическая и просодическая структуры речи

Минимальной смысловой единицей речи является *фонема*. В русском языке 42 фонемы, из них 6 гласных и 36 согласных. В других языках число фонем и соотношение гласных и согласных различно. Например, в английском языке 20 гласных (из них 5 дифтонгов) и 24 согласных, во французском — 16 гласных и 20 согласных.

[illegible]

Акустические характеристики каждой фонемы обусловлены артикуляционными особенностями ее образования: местом и способом. Под *местом образования* понимается положение сужений артикуляционного тракта в ядре фонем, определяющее его конфигурацию (артикуляционную статику) и в конечном итоге — его резонансные свойства. Место образования гласных фонем определяется положением тела языка (переднее-заднее, высокое-низкое) и положением губ (отгубленное-неотгубленное). Место образования согласных фонем определяется положением смычки или щели (губное, межзубное, зубное, альвеолярное, заальвеолярное, небное, фарингальное), а также задним или передним положением тела (мягкие и твердые).

группы фонем различаются в основном динамикой движения артикуляторов. Кроме того, взрывные, аффрикаты и щелевые дополнительно различаются по признаку наличия-отсутствия колебаний голосовых связок (звонкие-глухие).

Классификация фонем русской, английской и французской речи приведена на рис. 4.1. В плоскости «место согласных — способ» есть свободные клетки, которые в принципе могут быть заняты при рассмотрении фонемного состава других языков. Например, в состав фонем очень близкого к русскому белорусского языка входит ряд новых аффрикативных согласных, таких, как глухой твердый (ч), мягкий (ч') и звонкий мягкий (дз'). Подобные примеры можно было бы привести и для других языков. В плоскости «место согласных — способ» многие гласные не разделяются. Это связано с тем, что здесь не отражены степени огубленности, подъема и продвижения тела языка, которые для многих фонетических систем принимают более чем два значения, а также дополнительные характеристики способа образования гласных: монофтонги-дифтонги, краткие-длинные, чистые-незализованные.

В потоке речи в зависимости от конкретных условий акустико-артикуляционные характеристики фонем изменяются, что приводит к появлению оттенков фонем — *аллофонов*. Аллофоны подразделяются на комбинаторные и позиционные. *Комбинаторные аллофоны* обусловлены соседством данной фонемы с другими фонемами и образуются вследствие наложения в потоке речи артикуляции одного звука на другой — так называемого эффекта коартикуляции. Примерами коартикуляции является огубление согласных перед гласными (о) или (у), назализация гласных после мягких согласных и др. *Позиционные аллофоны* обусловлены положением фонемы в слове или фразе по отношению к ударному слогу, концу и началу слова и т. д. Например, редукция гласных в безударном положении, оглушение сонорных согласных (р), (л), (й) в конце фразы, удлинение шумовой фразы взрывных фонем в конце фразы (аффрикация) и др. Учет этих и других факторов позволяет дать оценку общему числу аллофонов гласных $N_g = 480$ и согласных $N_c = 8880$ которая, указывая порядок этого числа, наглядно иллюстрирует степень вариативности фонем в речевом потоке.

Другой класс лингвистических понятий, учет которых исключительно важен при создании высококачественных СРО, это интонация и ударение, определяющие *просодию речи*. Интонация и ударение играют важную роль при восприятии речи человеком. По ним определяются коммуникативная направленность высказывания, логический смысл, выделение главного и общего (рема и тема), осуществляется вычленение семантически связанных отрезков речи и объединение речевых элементов внутри этих отрезков. Зачастую в зависимости от интонационного оформления логически идентичные высказывания могут иметь разное семантическое значение. Физически интонация и ударение реализуются совокупностью акустических средств — просодических характеристик речи, к которым относятся:

мелодика — изменение частоты основного тона голоса;

ритмика — текущее изменение длительности звуков и пауз;

энергетика — текущее изменение интенсивности звука.

Процесс чтения текста человеком предполагает наличие автоматической процедуры формирования основного тона, интенсивности звука, длительности звуков и пауз на основе анализа входного текста. Это значит, что преобразование орфографического текста в последовательность фонем должно сопровождаться выделением информации, необходимой для задания просодических характеристик речевых сигналов. Для этой цели текст анализируется и по определенным правилам разбивается на следующие основные элементы: фраза, синтагма, акцентная группа, фонетическое слово. Эти элементы маркируются соответственно фразовым, синтагматическим, групповым и словесным ударениями. Каждой синтагме присваивается один из возможных интонационных типов. Основными интонационными типами являются неконечный, повествовательный, вопросительный. Существует также большое число модальных и эмоциональных оттенков этих основных интонационных типов.

Минимальным элементом, из совокупности которых складывается интонация синтагмы, фразы и текста в целом, является акцентная группа. Под *акцентной группой* понимается одно или несколько полнозначных слов синтагмы, объединенных общим просодическим (мелодическим, ритмическим и энергетическим) контуром, привязанным к единому групповому ударению. Число акцентных групп не всегда совпадает с числом фонетических слов в синтагме. Выделение слова в акцентной группе делает словесное ударение главным или основным. Другие слова в акцентной группе получают слабое или неосновное ударение. Основное ударение чаще всего получает первое слово акцентной группы. Это позволяет сформулировать простое правило установления границ акцентной группы: левая граница совпадает с началом слова, имеющего групповое ударение, правая — с началом слова следующей акцентной группы.

Акцентные группы ранжируются относительно границ синтагмы на конечные, начальные и срединные, которые вносят различный по значимости вклад в формирование просодического контура синтагмы. Основное разнообразие мелодических контуров реализуется на конечной акцентной группе, существенно меньшее — на начальной и срединной.

Мелодика, ритмика и энергетика акцентной группы задаются нормированными значениями частоты, длительности и интенсивности на трех ее участках: ядре, предъядре и заядре. *Ядром* акцентной группы является ударный слог, отмеченный знаком группового ударения, *предъядром* и *заядром* — соответственно предшествующие ему и следующие за ним фонемы акцентной группы.

Как показали экспериментальные исследования просодических контуров естественной речи и опыты по восприятию интонации и ударения синтетических стимулов, интонационный тип фразы определяется положением максимума мелодического контура относительно главноударного гласного. Положению максимума в центре гласного соответствует интонация побуждения или восклицания, сдвигу его влево — интонация завершения или утверждения, а сдвигу вправо — интонация незавершенности или вопроса. Другие параметры мелодического контура — значение максимума и его ширина — более важны при передаче эмоциональных оттенков. Положение ударения в слове, синтагме или фразе определяется в большей степени длительностью ударного гласного и в меньшей его интенсивностью. Положение ударения в слове узнается в 100 % случаев, если длительность ударного гласного превышает длительность безударного более чем в 1,5 раза.

4.3. Акустические характеристики фонем

Акустические характеристики фонем тесно связаны с артикуляционными особенностями их образования — местом и способом.

Речевой аппарат человека представляется в виде двух параллельных каналов — ротового и носового, образующих единую акустическую систему, возбуждаемую периодическими колебаниями голосовых связок либо турбулентным шумом. Распространение акустических волн в такой системе описывается уравнением Вебстера

$$\frac{1}{S(x)} \frac{\partial}{\partial x} \left(S(x) \frac{\partial p}{\partial x} \right) = \frac{1}{c^2} \frac{\partial^2 p}{\partial t^2},$$

где $S(x)$ — функция площади сечения тракта вдоль оси x распространения волн; p — давление; c — скорость звука; t — время.

Решение этого уравнения для речевого тракта различных конфигураций, соответствующих различным звукам речи, является основным предметом исследования акустической теории речеобразования. В результате решения получается аналитическое выражение для передаточной функции речевого тракта $H(p)$. В общем случае при достаточно малых потерях (что вполне справедливо для

реального речевого тракта)

$$H(p) = \frac{\prod_{m=1}^{\infty} (\tau_m^2 p^2 + 2\varepsilon_m \tau_m p + 1)}{\prod_{n=1}^{\infty} (\tau_n^2 p^2 + 2\varepsilon_n \tau_n p + 1)},$$

где p — комплексная переменная; $\tau_{n,m}$ и $\varepsilon_{n,m}$ — соответственно постоянная времени и коэффициент затухания.

Из этого выражения видно, что передаточная функция речевого тракта описывается произведением передаточных функций бесконечного числа резонансных звеньев второго порядка. Каждое n -е звено соответствует n -й форманте речевого сигнала, а m -е звено — m -й антиформанте. Частота и полоса n -й форманты определяются соответственно выражениями

$$F_n = 1/(2\pi\tau_n); B_n = \varepsilon_n/(\pi\tau_n).$$

На практике передаточная функция аппроксимируется ограниченным числом формант (до четырех) с помощью последовательно соединенных формантных фильтров.

Анализ вида передаточных функций речевого тракта для различных групп звуков показал, что достаточно полно описать их акустические характеристики можно, используя формантную модель (рис. 4.2). Управляющими параметрами модели являются следующие 10 формантных параметров: A_p , A_n — амплитуды голосового возбуждения ротовых и носовых формант; A_ϕ — амплитуда шумового возбуждения фрикативных формант; A_a — амплитуда аспиративного возбуждения ротовых формант; F_0 — частота основного тона; F_1 , F_2 , F_3 — частоты первой, второй и третьей формант; F_ϕ — частота фрикативных формант; B_ϕ — полоса пропускания фрикативных формант.

Приведенная формантная модель с успехом используется в системах синтеза речи, а ее управляющие параметры являются примером полного набора информативных признаков фонем для систем распознавания речи. Основную роль в разделении звуков речи по месту образования играют параметры F_2 , F_3 , F_ϕ , B_ϕ , а по способу образования — A_p , A_ϕ , A_n , F_0 , параметры F_1 и A_a участ-

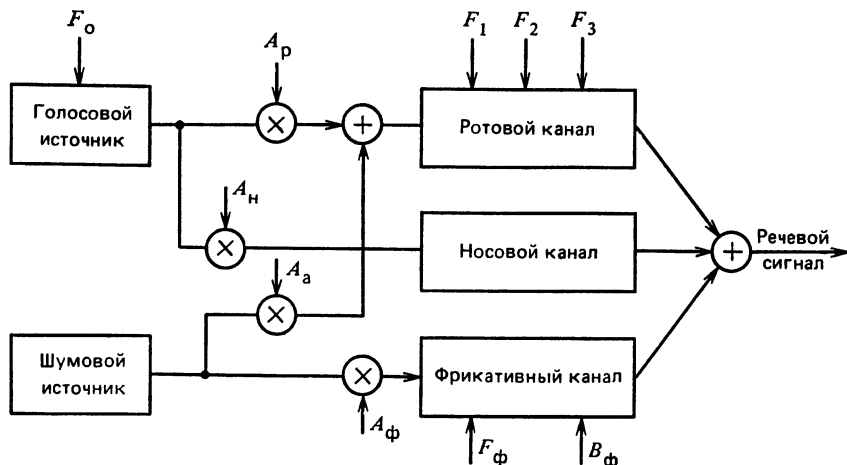


Рис. 4.2. Формантная модель акустики речевого тракта

Рис. 4.3. Зависимость формантных частот гласных русской и английской речи

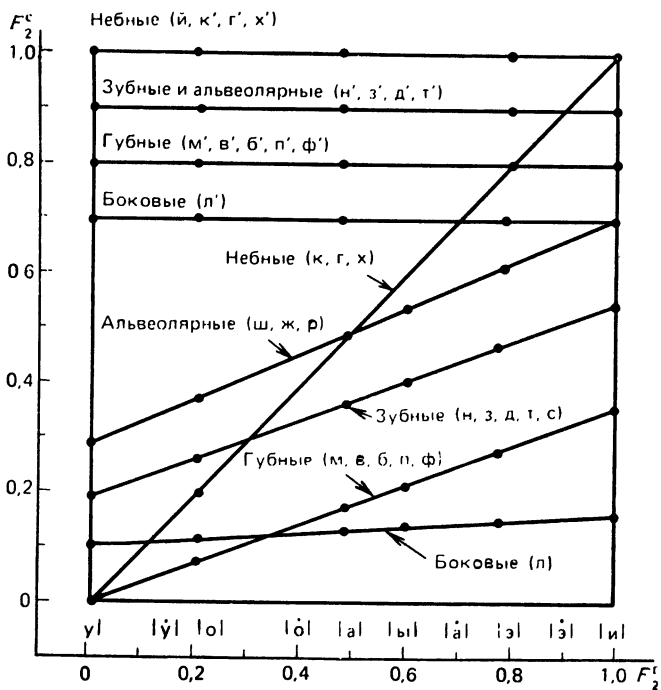
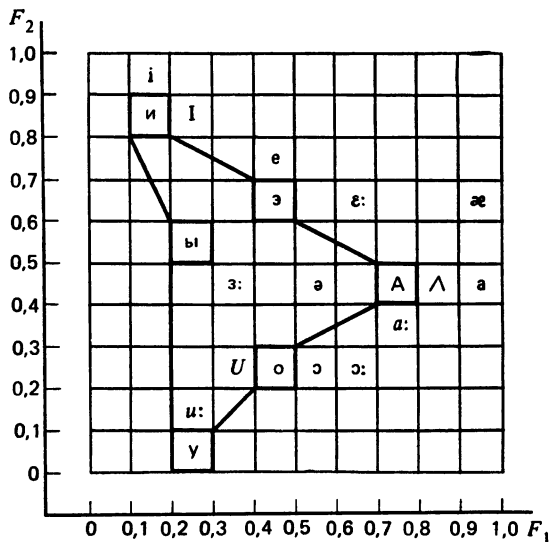


Рис. 4.4. Зависимость частоты второй форманты согласных от частоты гласных в слове

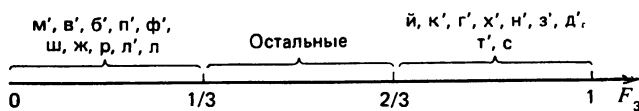


Рис. 4.5. Значения параметров F_3 , F_ϕ для различных согласных

а)



б)

вуют в равной степени в разделении как по месту, так и по способу образования.

Рассмотрим конкретную роль каждого формантного параметра в процессе разделения фонем, считая при этом, что параметры нормированы и принимают значения в интервале 0—1.

Группа гласных фонем отличается от согласных тем, что $A_p=1$, а $A_\phi=A_a=A_n=0$. Внутри группы, т. е. по месту образования, гласные разделяются значениями параметров F_1 , F_2 , F_3 , причем основную роль играют F_1 и F_2 (рис. 4.3).

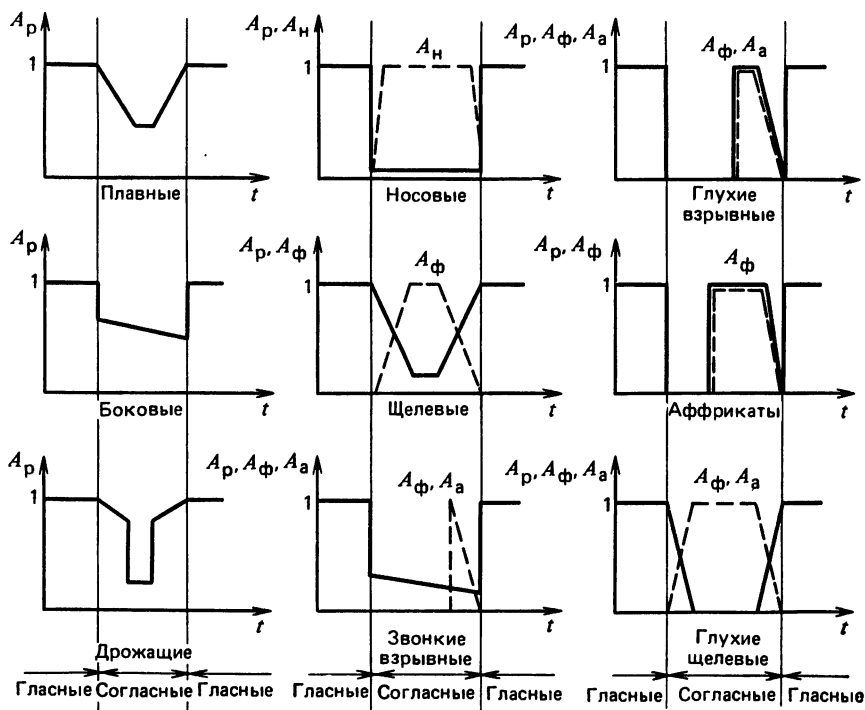


Рис. 4.6. Временные диаграммы изменения A-параметров для согласных разного способа образования

Наибольшую роль в разделении согласных по месту образования играет параметр F_2 . Причем этот параметр вследствие коартикуляции зависит от места образования не только согласного, но и окружающих его гласных (рис. 4.4). Наибольшему коартикуляционному воздействию подвергается группа небных твердых согласных (к, г, х), наименьшему — мягкие согласные.

Для дальнейшего разделения согласных по месту образования используются параметры F_3 , F_ϕ , V_ϕ , A_ϕ . Основную роль в разделении мягких согласных играет параметр F_3 (рис. 4.5,а), в разделении зубных и альвеолярных согласных — параметр F_ϕ (рис. 4.5,б). Дополнительное выделение группы губных согласных осуществляется по параметру V_ϕ , который равен 1 только для этой группы, а для остальных фонем равен 0. Аналогично параметр $A_\phi=1$ только для группы небных согласных и благодаря этому участвует в их дополнительном выделении.

В разделении фонем по способу образования важную роль играют значения параметров A_p , A_ϕ , A_a , A_n , F_1 , F_0 , определенные на участке ядра фонемы. Не менее важной является также временная организация этих параметров, которая в значительной степени определяется способом образования фонем. На временных диаграммах рис. 4.6 отражены значения параметров A_p , A_n , A_ϕ , A_a , характеризующие ту или иную группу согласных, а также временные особенности их развития в контексте гласный — согласный — гласный.

4.4. Информационная структура речевого сигнала

Трудности в распознавании речевых образов связаны с чрезвычайной изменчивостью основных характеристик речевого сигнала. Отсутствие законченной модели сигнала не позволяет грамотно сформулировать требования к первичным анализирующим устройствам. В результате решения, принимаемые на более высоких уровнях распознавания, сопровождаются большим количеством ошибок.

Ниже требования к структурной схеме устройства для первичного анализа будут сформулированы на основе представления о речевом сигнале как о сигнале в специфической «речевой» системе связи. При таком подходе речевой сигнал рассматривается не в отрыве от всего процесса общения, а как следствие речевой связи между двумя или более говорящими.

Как известно из теории связи, для передачи любых сообщений требуется вполне определенная структура системы связи. Всякая техническая система связи включает такие функциональные элементы, как источники информации, кодирующие устройства, переносчики, модуляторы, системы разделения каналов, демодуляторы, декодеры и др. Естественные системы связи, в частности устная речь, не являются в этом смысле исключением. Основной особенностью речевого сигнала, вытекающей из такого рассмотрения, является его полиинформативность и полимодуляционность.

Генерирование речевого сигнала связано с функционированием трех источников звука — переносчиков: тонального, шумового и импульсного (рис. 4.7). Передача информации осуществляется путем модуляции некоторых параметров α , β , γ , ... каждого из указанных переносчиков.

Тональный переносчик в отсутствие модуляции может быть описан как периодическое колебание $y(t)$, создаваемое движением голосовых связок, со спектром

$$y(t) = A \sum_{k=1}^n a_k \cos(k\omega_1 + \varphi_k),$$

где A — среднеквадратическое значение амплитуд спектральных составляющих; a_k — нормированные амплитуды k -х гармоник; ω_1 — частота первой гармоники; φ_k — фазовые сдвиги k -х гармоник; n — число гармоник. В процессе речеобразования модулируемыми параметрами могут быть амплитуда A — амплитудная модуляция, $\{a_k\}$ — модуляция формы спектра и фаза $\{\varphi_k\}$ — фазовая модуляция спектра.

Шумовой переносчик в отсутствие модуляции может быть описан как гауссовский стационарный процесс, исчерпывающей характеристикой которого являет-

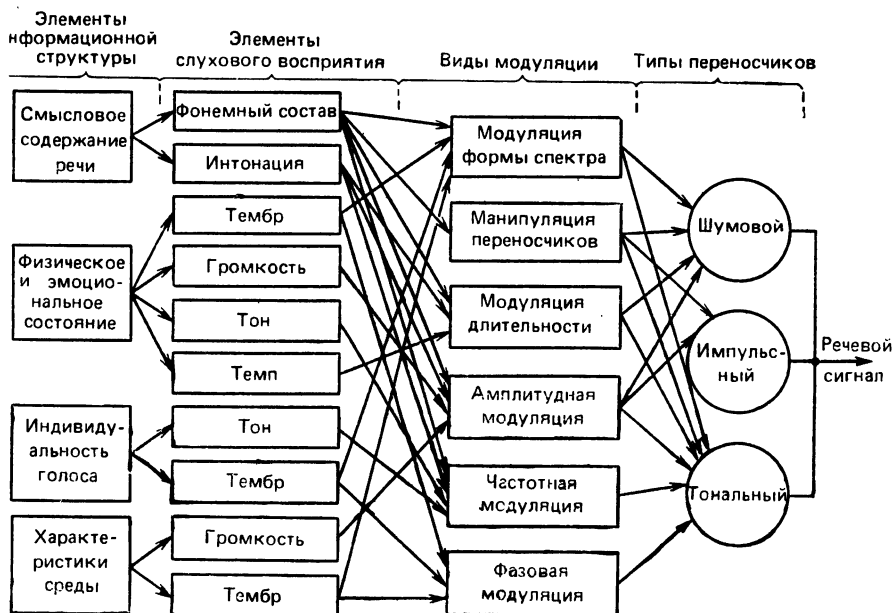


Рис. 4.7. Схема формирования информационной структуры речевого сигнала

ся его функция корреляции или спектр мощности

$$R(\tau) = P \sum_{k=1}^n d_k \cos k\Omega\tau,$$

где $\Omega = \pi/T$; T — интервал корреляции; d_k — коэффициенты разложения нормированной функции корреляции в ряд Фурье; P — мощность случайного процесса. В процессе речеобразования основными модулируемыми параметрами являются P — амплитудная модуляция и $\{d_k\}$ — модуляция формы спектра.

Импульсный переносчик характеризуется резким нарастанием скорости воздушного потока с дальнейшим экспоненциальным его спадом при раскрытии смычки у взрывных звуков [Фант, 1964]:

$$I(t) = \begin{cases} U \exp [\alpha(t - \tau_0)] & \text{при } t > \tau_0, \\ U & \text{при } t \leq \tau_0. \end{cases}$$

Основным модулирующим параметром является U , который характеризует на пряженность взрывного звука, — амплитудная модуляция.

Кроме указанных видов модуляции целесообразно выделить в отдельный класс модуляцию длительности звуков и манипуляцию переносчиков как предельный случай амплитудной модуляции, характеризуемой дискретным процессом их независимого включения или выключения.

Как видно из рис. 4.7, наибольшее число видов модуляции, а следовательно, и наибольшую информационную емкость имеет тональный переносчик, а наименьшую — импульсный. Роль каждого вида модуляции в передаче элементов информационной структуры речевого сигнала поясним на примере передачи информации о фонемном составе и интонации речи.

При передаче информации о фонемном составе речи осуществляется непрерывный процесс перестройки речевого тракта. Это приводит к непрерывному

изменению формы мгновенных спектров речевого сигнала — модуляция формы спектров — и его средней мощности — амплитудная модуляция. При перестройке речевого тракта осуществляется также фазовая модуляция, хотя полной ясности ее роли в восприятии речевых звуков нет. Кроме того, процесс передачи информации о фонемном составе связан с постоянной сменой комбинаций включения переносчиков — манипуляция переносчиков — и с изменением частоты основного тона на смычках звонких взрывных звуков — частотная модуляция. Информация об интонационной окраске речи, а также о физическом и эмоциональном состоянии, об индивидуальных особенностях голоса и характеристике электроакустической среды передается в основном путем модуляций частоты основного тона — частотная модуляция — и общего уровня звуков — амплитудная модуляция.

Рассмотренная схема передающего устройства речевой системы связи не претендует, конечно, на полноту охвата всех возможных видов передаваемой и получаемой информации, а также всего разнообразия видов модуляции. На схеме рис. 4.7 не указана такая важная операция, непосредственно предшествующая модуляции переносчиков, как кодирование передаваемых сообщений. Однако в целом из рассмотрения схемы на рис. 4.7 можно сделать несколько важных выводов:

в процессе речевого общения осуществляется параллельная передача различных видов информации;

для передачи информации каждого вида одновременно используются несколько видов модуляции;

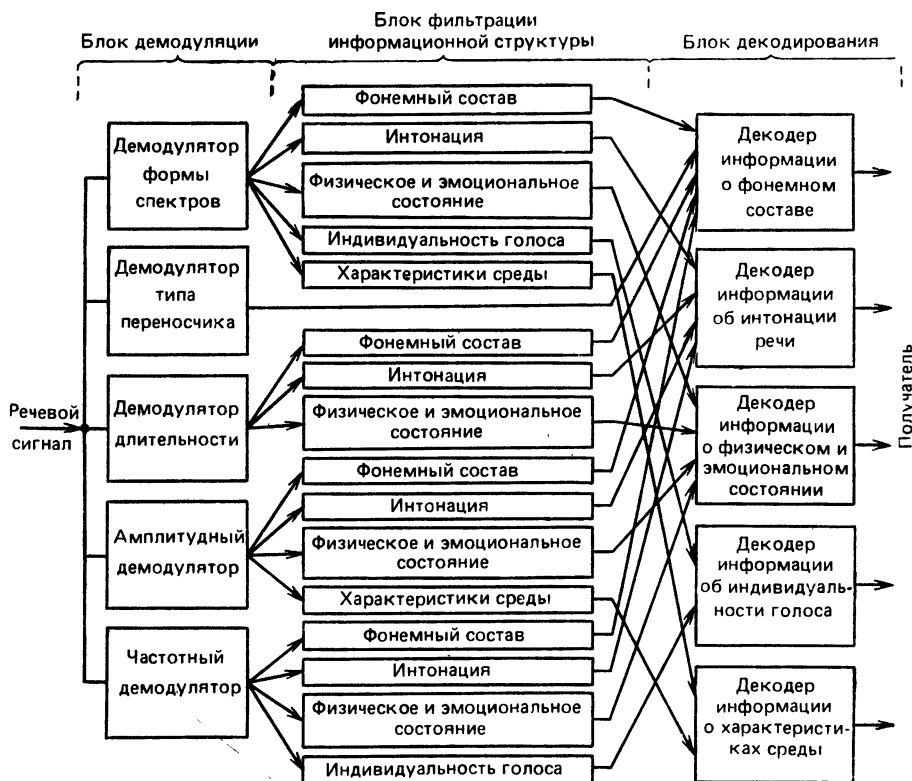


Рис. 4.8. Схема приемного устройства «речевой» системы связи

ни один из видов модуляции (кроме манипуляции переносчиков) не используется только для передачи одного вида информации.

Учитывая указанные особенности формирования речевого сигнала, можно сформулировать вполне определенные требования к схеме его обработки в приемном устройстве (анализаторе).

1. При анализе заданного элемента информационной структуры должна осуществляться демодуляция (детектирование) речевого сигнала по каждому виду модуляции, посредством которой ведется его передача.

2. Результат детектирования по каждому виду модуляции должен быть инвариантен относительно остальных видов модуляции. Возможным методом достижения такого рода инвариантности является осуществление предварительной нормализации речевого сигнала.

3. Если с помощью данного вида модуляции одновременно осуществляется передача еще из других элементов информационной структуры, то полученный в результате демодуляции сигнал должен быть подвергнут дальнейшим операциям разделения.

На рис. 4.8 приведена гипотетическая схема приемного устройства речевой системы связи, построенная в соответствии с указанными требованиями. Речевой сигнал одновременно поступает на входы блока демодуляции. Блок демодуляции состоит из собственно демодулятора (детектора данного вида модуляции) и нормализатора по основным видам модуляции. С выходов демодуляторов медленно меняющиеся сигналы (модулирующие) поступают на блок фильтрации элементов информационной структуры. Осуществление фильтрации возможно благодаря разнесу по частотным диапазонам, используемым при передаче различных видов информации. Так, если рассмотреть информационный состав результатов демодуляции формы спектра, то оказывается, что информация о фоне в составе заключена в сравнительно быстрых изменениях спектра (от звука к звуку), информация об эмоциональном и физическом состоянии говорящего — в значительно более медленных (от состояния к состоянию). Еще более медленные изменения спектра связаны с передачей информации об индивидуальных особенностях голоса диктора (от диктора к диктору) и о характеристиках среды. Таким образом, предполагается существование следующей цепочки неравенств:

$$\Omega_{\text{фон}}^{\text{max}} > \Omega_{\text{фон}}^{\text{min}} > \Omega_{\text{max}}^{\text{SM}} > \Omega_{\text{min}}^{\text{SM}} > \Omega_{\text{max}}^{\text{D}} > \Omega_{\text{min}}^{\text{D}} > \dots$$

где Ω_{max} , Ω_{min} — частотные диапазоны изменений модулируемого параметра при передаче каждого элемента информационной структуры.

Полученные на выходе блока фильтрации непрерывные, медленно меняющиеся параметры используются затем как входные сигналы для блоков декодирования. Эти параметры образуют, во-первых, полную систему для любого вида информации, и, во-вторых, их значения не зависят от изменений в речевом сигнале, связанных с передачей других видов изменений. Получение полной системы таких параметров и является основной задачей первичного анализа речевых сигналов.

Глава 5.

Принципы построения систем речевого общения

Б. М. Лобанов

5.1. Классификация речевых процессоров

В общем случае СРО строятся на базе специализированных речевых процессоров двух основных типов: анализаторов и синтезаторов.

5.1.1. Анализаторы. Эти устройства предназначены для преобразования речевых сигналов с микрофона (информационный поток сигналов 10^5 бит/с) в

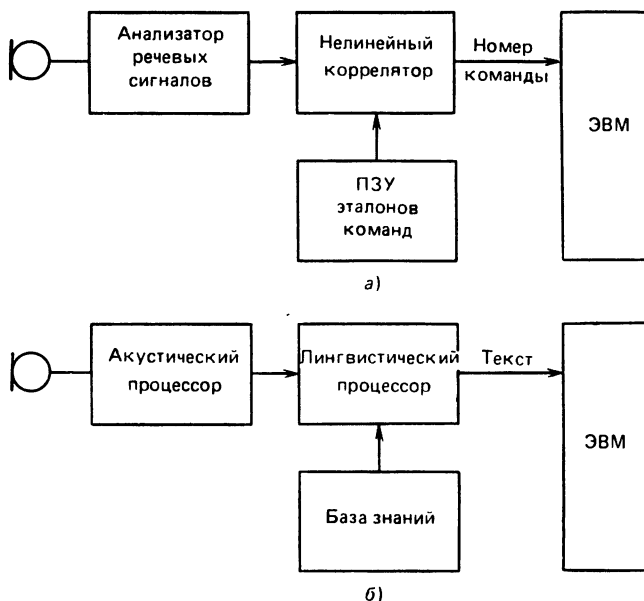


Рис. 5.1. Схемы анализаторов речевых сообщений ограниченного словаря (а) и универсального (б)

последовательность цифровых кодов с существенно меньшим информационным потоком (10^4 — 10^5 бит/с) и с обязательным сохранением передачи смыслового компонента речи. Анализаторы подразделяются на два основных класса: анализаторы сигналов и анализаторы сообщений.

В анализаторах сигналов сокращение информационного потока достигается только за счет учета акустических и статистических характеристик речевого сигнала без обращения к его смысловой функции. Примерами анализаторов сигналов являются Δ -кодеки, спектрально-полосные, анализаторы линейного предиктивного кодирования (ЛПК-анализаторы), формантные [Сапожков и др., 1983].

В анализаторах речевых сообщений (распознавателях) осуществляется сжатие информационного потока за счет введения операции распознавания смысловых элементов речи (фразы, слова, морфемы, фонемы) [Златоустова и др., 1986]. Анализаторы речевых сообщений, в свою очередь, подразделяются на две группы: анализаторы ограниченного словаря и универсальные.

Анализаторы ограниченного словаря ориентированы на распознавание заданного конкретной задачей числа речевых команд (обычно порядка 10^2), т. е. на идентификацию одной из произнесенных речевых команд словаря в виде номера команды (рис. 5.1,а). Распознавание осуществляется путем нелинейного во времени сопоставления эталонов команд с произносимой командой и выбора наиболее схожего эталона. В большинстве существующих анализаторов ограниченного словаря формирование эталонов осуществляется в процессе обучения на используемый словарь команд и голос диктора. Чаще всего процесс обучения состоит в однократном прочтении оператором всего словаря команд. Еще одним ограничением большинства современных анализаторов этого типа является требование изолированного произнесения речевых команд, т. е. с паузами между словами от 0,3 до 1 с. Распознавание слитной речи даже ограниченного словаря — пока нерешенная научная проблема.

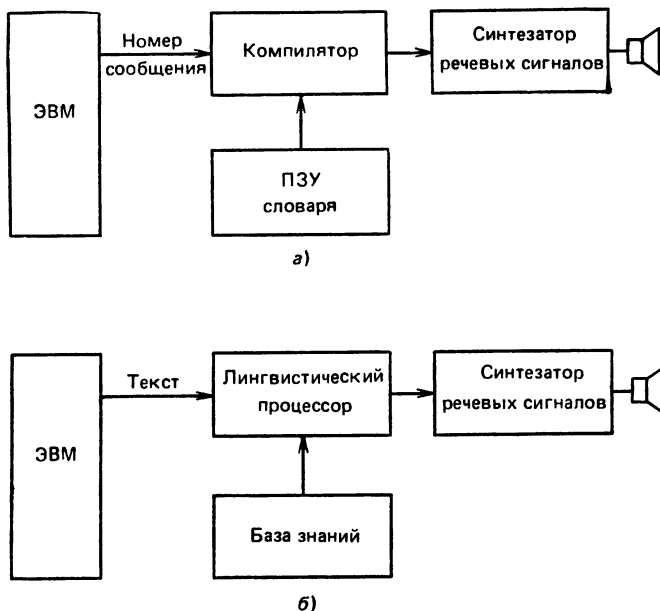


Рис. 5.2. Схемы синтезаторов речевых сообщений ограниченного словаря (а) и универсального (б)

Универсальные анализаторы ориентированы на текущее распознавание полного набора смысловых элементов речи (фонем или морфем), из которых может быть составлено и в конечном счете распознано любое слово или слитно произнесенное речевое сообщение (рис. 5.1,б). Распознавание осуществляется лингвистическим процессором по правилам, заложенным в базе знаний.

5.1.2. Синтезаторы речи. Эти устройства предназначены для преобразования кодовой последовательности, поступающей от ЭВМ, ПЗУ или линии связи, в непрерывный речевой сигнал. Синтезаторы подразделяются на классы и группы по тем же признакам, что и анализаторы речи. Классам анализаторов речевых сигналов и сообщений соответствуют синтезаторы речевых сигналов и сообщений.

Синтезаторы сообщений делятся по аналогичным признакам на синтезаторы ограниченного словаря — компиляторы и универсальные.

В *компиляторах* (рис. 5.2,а) любое сложное речевое сообщение может быть получено путем компиляции (простого соединения) элементов речи. Элементы речи начитываются диктором, соответствующие им сигналы компрессируются тем или иным способом, кодируются и записываются в ПЗУ. При синтезе речевого сообщения из ПЗУ закодированные речевые элементы считываются в нужной последовательности и одновременно восстанавливается речевой сигнал. Очевидной простотой компиляционного метода и его технической реализации объясняется большое количество сообщений о проектируемых и законченных разработках компиляторов и их применении.

Удовлетворительный по качеству компиляционный синтез речи возможен лишь при использовании в качестве элементов речи отдельных фраз либо словоформ, подставляемых в определенное место стандартной фразы. Попытки добиться высококачественного синтеза произвольного текста простой компиляцией словоформ, слогов или аллофонов не привели к положительным результатам.

Характеристики синтезаторов

Параметр	Компильбатор					Универсальный
	ИКМ	Δ-кодек	полосный	ЛПК	формантный	
V , бит/с	$64 \cdot 10^3$	$16 \cdot 10^3$	$8 \cdot 10^3$	$2 \cdot 10^3$	$1 \cdot 10^3$	64
N , слов	1	4	8	32	64	1000

Все эти элементы речи тесно связаны внутри фразы. В слитной речи не существует аналогов этих элементов, произнесенных изолированно, и наоборот, речь из изолированно произнесенных элементов звучит ненатурально. В связи с этим подготовка словаря в компильбаторах представляет собой самостоятельную и сложную проблему. Для успешного ее решения в каждом конкретном случае применения синтезатора многие фирмы за рубежом идут на создание специализированных центров проектирования речи, оснащенных соответствующим оборудованием и персоналом лингвистов-прикладников.

При разработке *универсальных синтезаторов* речевых сообщений стремятся получить функциональную модель речеобразования, адекватную реально существующим языковым и акустическим явлениям. На входе такой модели орфографический или фонемный текст произвольного содержания, на выходе — звучащая речь (рис. 5.2,б). По своему существу разрабатываемые в рамках данного подхода синтезаторы являются кибернетической функциональной моделью чтения текста человеком. В базе знаний синтезатора хранится не только информация об элементарных единицах речи (эталон фонем и интоном), но и алгоритмические правила их модификации в зависимости от конкретного контекста звуковой реализации. Процесс применения этих правил к эталонам фонем и интоном для входного синтезируемого текста реализуется лингвистическим процессором. К настоящему времени качество речи и стоимость универсальных синтезаторов достигли коммерчески приемлемых показателей, и они начинают оказывать серьезную конкуренцию компильбаторам в силу простоты применения, малого расхода памяти на элемент речи, неограниченности состава словаря синтезируемых сообщений. (Основы теории универсальных синтезаторов изложены в работах [Кучеров и др., 1983; Лобанов, 1988].)

Одним из основных параметров, характеризующих синтезаторы речи, является емкость памяти V , необходимая для хранения 1 с речи, — примерно 1,5—2 слова. Для компильбаторов она определяется типом параметрического представления речевого сигнала, для универсальных синтезаторов не зависит от типа используемого синтезатора речевых сигналов, а определяется числом фонем (букв текста), произносимых в 1 с (табл. 5.1, где N — число слов длительностью 0,5 с, записанных в ПЗУ емкостью 4 Кбайт).

В классе компильбаторов выделены наиболее распространенные типы, различающиеся методом представления речевого сигнала: ИКМ — импульсно-кодовая модуляция; Δ-кодек — дельта-модуляция; полосный — спектрально-полосное; ЛПК — линейное предиктивное кодирование; формантный. (Описание принципов работы анализаторов-синтезаторов: ИКМ, Δ-кодек, полосных и ЛПК — читатель может найти в монографиях [Сапожков, 1963; Сапожков и др, 1983].)

5.2. Формантный анализатор речевых сигналов

Под *формантами* в общем случае понимают полосы передаточной функции речевого тракта, характеризующиеся частотой F_i , амплитудой A_i и полосой пропускания B_i . Обычно интересуются первыми тремя-четырьмя наиболее информативными формантами и их двумя наиболее информативными параметрами F_i и A_i . На амплитудно-частотной характеристике (АЧХ) форманты проявля-

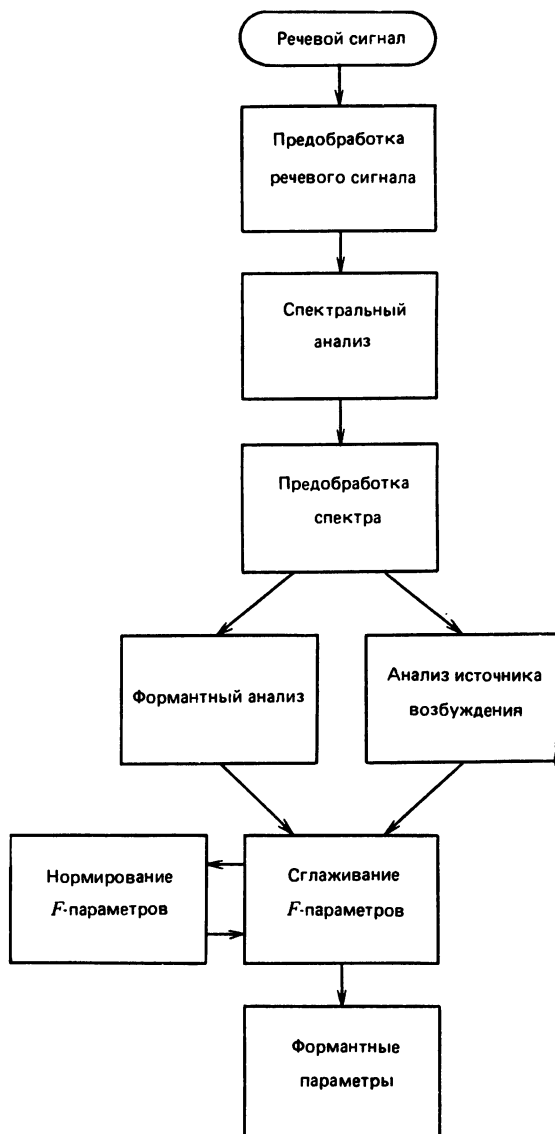


Рис. 5.3. Общая схема формантного анализа речевых сигналов

блоке формантного анализа. В этом же блоке осуществляется детектирование сигналов корректора F_0 , F_1 , используемых затем в блоке анализа источника голосового возбуждения. Сформированные таким образом сигналы через коммутатор подаются на вход аналого-цифрового преобразователя (АЦП).

В блоке спектрального анализа (рис. 5.4,б) преобразованный в АЦП сигнал

ются в виде заметных максимумов. Однако не всякий максимум является формантой. Обычно в диапазоне частот первых четырех формант ряд побочных максимумов не является формантами. Чтобы осуществить селекцию формантных максимумов, необходимо активно использовать всю имеющуюся априорную информацию о формантах.

Ниже описывается эффективный алгоритм измерения формантных параметров спектра, основанный на активном использовании априорной информации о соотношениях частот и амплитуд, а также о динамических свойствах формантных траекторий [Бухтилов и др., 1986]. Кроме того, определяются частота F_0 и амплитуда A_0 сигнала источника голосового возбуждения [Бухтилов и др., 1970]. Комплекс получаемых формантных параметров $\{F_i, A_i\}$ ($i=0-4$) достаточно полный для восстановления параметров формантной модели речевого тракта (см. § 4.3).

Формантный анализатор представляет собой аппаратно-программный комплекс (рис. 5.3).

В блоке предобработки речевого сигнала (рис. 5.4,а) осуществляется разделение речевого сигнала с помощью частотных корректоров, сжатие динамического диапазона каждого сигнала и их суммирование. Этим достигается выравнивание в среднем по уровню глобальных максимумов спектра в областях частот F_0 , F_1 и F_2-F_4 суммарного сигнала, используемого в

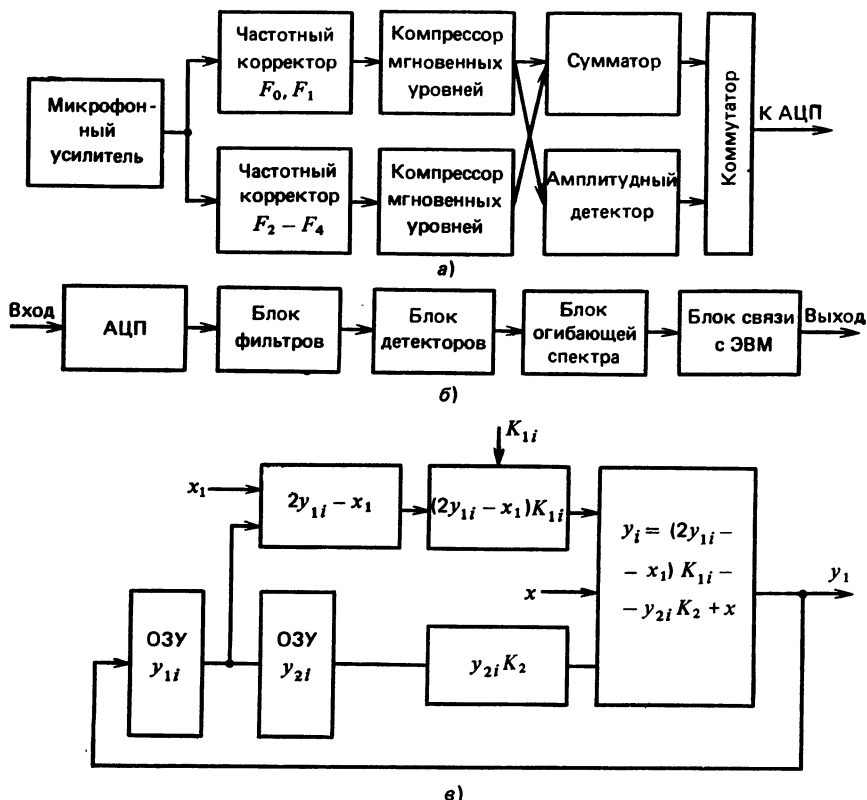


Рис. 5.4. Схемы блока предобработки речевого сигнала (а), спектрального анализа (б) и блока фильтров (в)

подвергается цифровой фильтрации четырехполосными фильтрами (рис. 5.4, в). Блок фильтров дважды выполняет вычисление 64 функций

$$y_i = (2y_{1i} - x_1)K_{1i} - y_{2i}K_2 + x, \quad i = 1-64,$$

где x , x_1 — входной сигнал в момент t и задержанный на один период дискретизации; y_i , y_{1i} , y_{2i} — результат фильтрации i -го спектрального канала и его задержанные значения; K_2 — коэффициент, общий для всех фильтров, определяющий их полосу частот; K_{1i} — 64 коэффициента, определяющие частоты настройки фильтров.

В блоках детекторов из огибающей спектра на интервалах длительностью 1 мс выделяется максимум продетектированного сигнала в каждом из 64 каналов; 48 спектральных каналов используются для формантного анализа; 16 узкополосных в диапазоне частоты основного тона — для анализа F_0 и A_0 источников голосового возбуждения. Блок связи с ЭВМ содержит буферное ОЗУ для хранения-выдачи 64 8-разрядных выходных данных и узел связи с ЭВМ.

Схема алгоритма работы блока предобработки спектра приведена на рис. 5.5. Из усредненных в окне 10 мс спектральных данных формируется массив чисел для оценки параметра речь/пауза (межфразовая). В паузе, оцениваемой по пороговому значению энергии, измеряется уровень помех a_{in} во всех

64 спектральных каналах. Для последующей обработки формируются спектральные отсчеты:

$$a_i = a_{ic} - a_{ip},$$

где a_{ic} — текущие значения.

Схема алгоритма работы блока формантного анализа приведена на рис. 5.6. В блоке спектрального обострения формируются отсчеты

$$a_{i0} = (a_i + a_{i-1} + a_{i+1}) - (a_{i-2} + a_{i+2}), \quad i = 1-48,$$

позволяющие усилить контраст формантных максимумов спектра. В блоке выделения локальных максимумов спектр разбивается на четыре перекрывающиеся формантные зоны, в каждой из которых выделяются по три наибольших локальных максимума, характеризуемых частотой F_{ij} и амплитудой A_{ij} ($i = 1-4$, $j = 1-3$), 12 пар величин F_{ij} , A_{ij} являются претендентами на формантные частоты и их амплитуды.

Анализ претендентов на форманты производится в следующих блоках. В блоке селекции по амплитудам формируются величины

$$S_{ijkl} = A_{i1} + A_{2j} + A_{3k} + A_{4l}, \quad i, j, k, l = 1-3,$$

и ранжируются по значению. Задача выделения формант формулируется как поиск экстремума S_{ijkl} при ограничениях на F_{ij} и A_{ij} , рассматриваемых ниже. Блок селекции по частотам отсеивает комбинации претендентов, не удовлетворяющих условиям взаимного расположения формант:

$$(F_{2j} - F_{1j}) > a_1,$$

$$(F_{3j} - F_{2j}) > a_2,$$

$$(F_{4j} - F_{3j}) > a_3,$$

$$(F_{1j} + F_{2j}) < a_4,$$

$$(F_{2j} + F_{3j}) < a_5,$$

где a_1-a_3 и a_4, a_5 — константы, выражающие минимально и максимально допустимые расстояния по частоте между претендентами. В блоке селекции по динамическим характеристикам производится сопоставление претендентов по времени в прямом и обратном направлениях. Усиливаются амплитуды тех претендентов, частоты которых являются линейным продолжением частот $F_{t-2,i}$, $F_{t-1,i}$ в прямом направлении и $F_{t+1,i}$, $F_{t+2,i}$ в обратном.

После этого вырабатываются окончательные значения частот F_i и амплитуд A_i формант. Анализ параметров источника голосового возбуждения F_0 и A_0 осуществляется с помощью 16 узкополосных ($B_0 = 28$ Гц) фильтров в диапазоне изменения частоты основного тона $F_0 = 80-400$ Гц по огибающей речевого сигнала в этом диапазоне. Работа блока оценки параметров F_0 , A_0 основана на реализации алгоритма различения спектров периодического (дискретного) и шумового (непрерывного).

Сущность алгоритма заключается в нахождении суммы спектральных отсчетов на частотах гармоник основного тона сигнала и вычитании из нее суммы спектральных отсчетов на частотах между гармониками сигнала. Для различных дискретных значений F_{0i} ($i = 1-16$) составляется конечное число разностей указанных сумм S_{ij} , где i — номер дискретного значения частоты F_0 ; j — номер разности сумм. Для нахождения F_0 и A_0 определяют $\max_j S_{ij}$. Значение этого максимума принимается за A_0 , а приписанное ему значение i — за номер дискретного значения F_0 .

В блоке сглаживания (см. рис. 5.3) осуществляется устранение одиночных сбоев в выделении формантных частот и амплитуд, а также интерполяция траекторий формантных частот на неозвученных интервалах. Устранение одиночных сбоев достигается с помощью процедуры медианного сглаживания, а интерполяция — соединением участков траектории до и после вокальной паузы отрезком прямой линии.

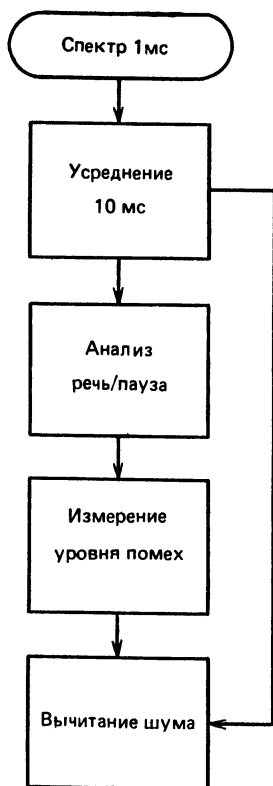


Рис. 5.5. Схема алгоритма предобработки спектров

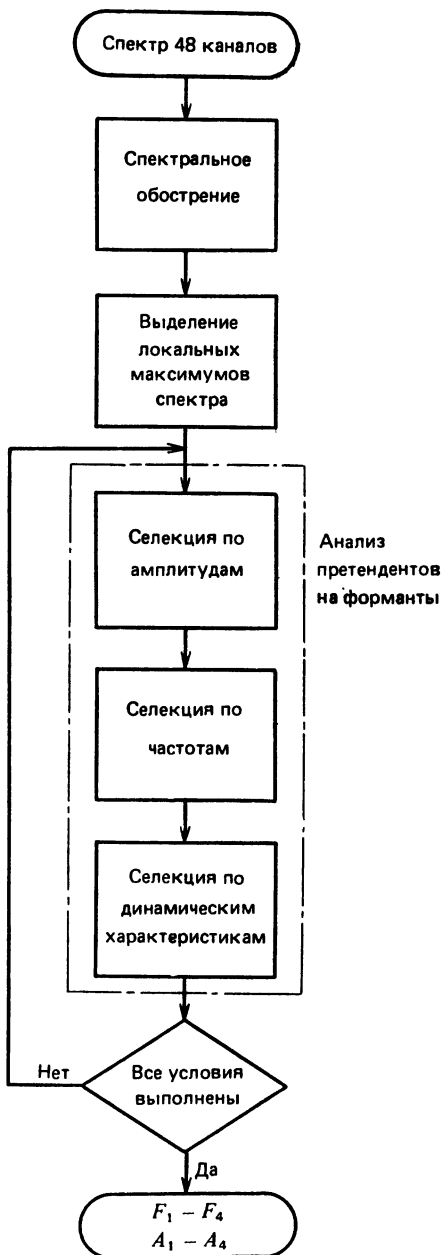


Рис. 5.6. Схема алгоритмов формантного анализа спектра

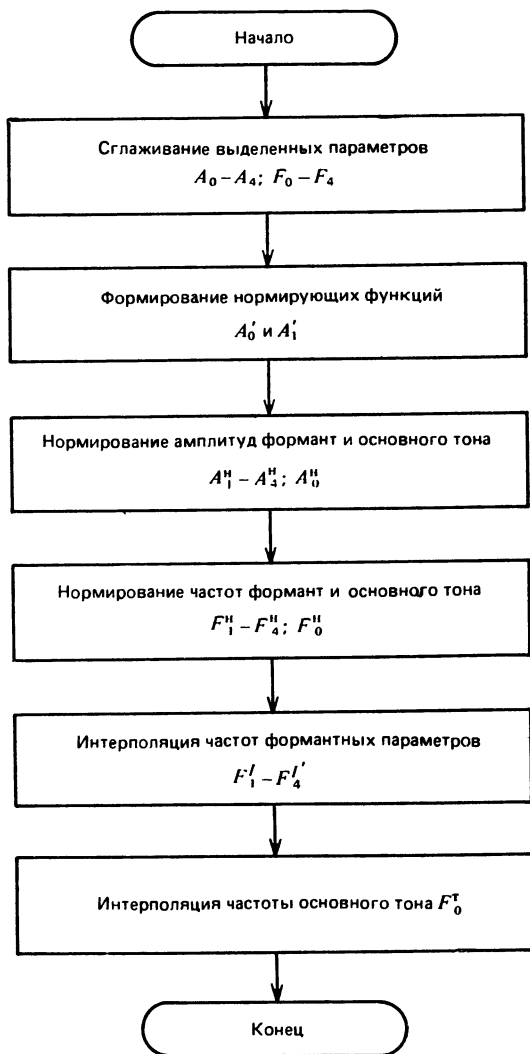


Рис. 5.7. Схема алгоритма сглаживания и нормирования параметров формант

Нормирование формантных амплитуд осуществляется с помощью нормирующих функций $\{A_i^n(t)\}$, формируемых из исходных функций $\{A_i(t)\}$ методом пикового детектирования их огибающей:

$$A_i^n(t) = A_i(t) / A_i^n(t).$$

Нормирование траекторий формантных частот F_i осуществляется по формуле

$$F_i^n(t) = (F_i(t) - F_{i\min}) / (F_{i\max} - F_{i\min}),$$

где $F_{i\max}$ и $F_{i\min}$ находятся по предварительно произнесенной обучающей фразе.

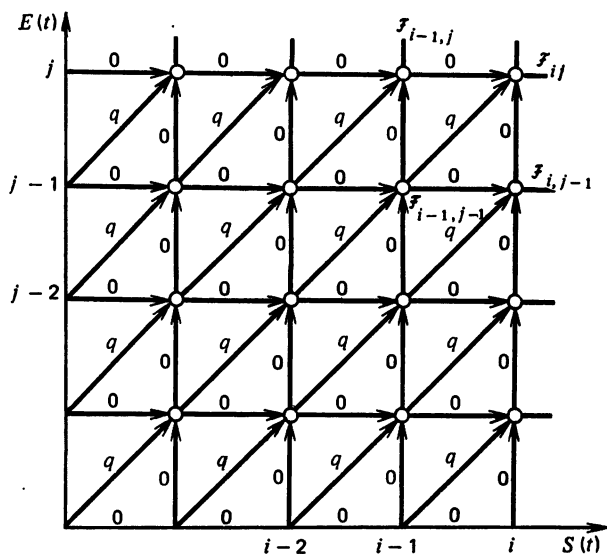
Схема алгоритма сглаживания и нормировки формантных параметров приведена на рис. 5.7.

5.3. ДП-анализатор речевых команд

В настоящее время еще преждевременно говорить о принципах работы универсальных анализаторов речевых сообщений — анализаторов, обеспечивающих распознавание слитной речи неограниченного словаря. По-видимому, создание эффективно действующего устройства типа пишущей с голоса машинки станет возможным не ранее 90-х годов. Сейчас же реально говорить только об устройствах для надежного распознавания нескольких сотен слов, причем на способ пользования ими налагается ряд довольно жестких условий. Прежде всего, слова должны произноситься в виде речевой команды —

слова-фразы с обязательной паузой. Во-вторых, требуется предварительная настройка анализатора на словарь и голос диктора путем как минимум однократного прочтения всего словаря речевых команд. В-третьих, в помещении недопустим высокий уровень шума, в противном случае пользователю необходимо применять шумозащищенный головной микрофон. При удовлетворении этих условий современные устройства обеспечивают надежность распознавания речевых команд порядка 99 %, что вполне приемлемо для многих применений.

Рис. 5.8. Граф-схема ДП-сравнения реализации с эталоном



Надежность распознавания порядка 99 % была достигнута главным образом благодаря использованию метода динамического программирования (ДП-метода). Предпосылкой создания ДП-метода [Виницук, 1987; Слуцкер, 1968] было стремление нормализовать временные деформации произносимой команды, возникающие вследствие произвольных изменений темпа и манеры произнесения. В результате одни звуки слова могут быть сильно растянуты, а другие сжаты до полной редукции. ДП-метод, применяемый при сравнении входной и эталонной реализаций слов, — это эффективная вычислительная процедура для нахождения наилучшего временного сопоставления эталона и реализации слова. Различия между реализацией и эталоном могут быть обусловлены либо принадлежностью их к разным классам, либо в случае принадлежности одному классу нелинейной деформацией временной шкалы. ДП-метод позволяет исключить влияние временного несоответствия между сравниваемыми сигналами на меру физической близости между ними.

Для распознавания речевых команд в памяти распознающего устройства должны храниться эталоны $E^k(t)$ всех k -х распознаваемых слов. Неизвестное слово сравнивается с каждым из эталонных путем расчета интегральной меры подобия $\mathcal{F}^k$. Эта мера рассчитывается исходя из значений локальных мер подобия q между дискретными элементами $E^k(t)$ и $S(t)$. Неизвестное слово $S(t)$ относится к тому эталону, на который оно больше всего похоже, т. е. мера подобия $\mathcal{F}^k$ максимальна.

Расчет интегральной меры подобия $\mathcal{F}^k$ сводится к определению оптимального пути на графе (рис. 5.8) с помощью рекуррентной формулы динамического программирования

$$\mathcal{F}_{ij}^k = \max [\mathcal{F}_{i-1,j}^k; \mathcal{F}_{i,j-1}^k; \mathcal{F}_{i-1,j-1}^k + q_{ij}],$$

где $\mathcal{F}_{ij}^k$ — интегральная мера подобия в вершине ij графа; i, j — номера дискретных элементов неизвестного слова $S(t)$ и эталона $E^k(t)$ соответственно; q_{ij} — локальная мера подобия между i -м элементом $S(t)$ и j -м элементом $E^k(t)$.

Если одному элементу эталона $E^k(t)$ соответствует несколько элементов неизвестного слова $S(t)$ (например, диктор произнес «восемь»), то на графе

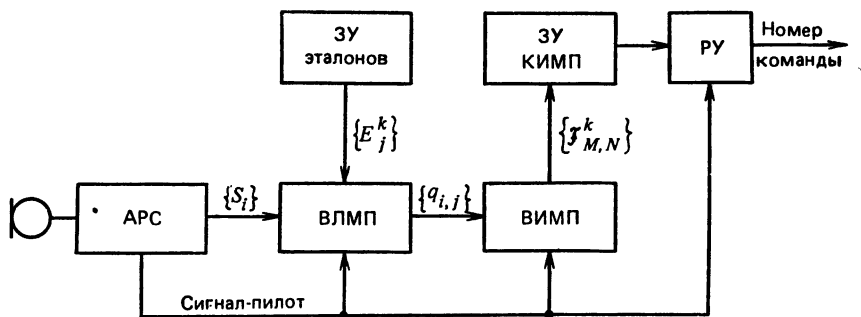


Рис. 5.9. Схема устройства распознавания речевых команд ДП-методом

оптимальный путь пройдет по горизонтали. Если одному элементу $S(t)$ соответствует несколько элементов $E^k(t)$, то оптимальный путь пройдет по вертикали. Простое удлинение некоторых участков речи не ведет к изменению интегральной меры подобия $\mathcal{F}^k$, поэтому вертикальные и горизонтальные линии графа имеют вес, равный нулю. Увеличение интегральной меры подобия на $q_{i,j}$ происходит только при прохождении оптимального пути по диагонали.

Схема устройства распознавания речевых команд ДП-методом изображена на рис. 5.9. С выхода анализатора речевых сигналов (АРС) и ЗУ эталонов речевых команд в вычислитель локальных мер подобия (ВЛМП) поступают отсчеты $\{S_i\}$ и $\{E_j^k\}$. Начало работы ВЛМП инициируется сигналом-пилотом, вырабатываемым АРС и указывающим начало ($i=0$) и конец ($i=M$) произносимой речевой команды. Для каждого дискретного отсчета времени ВЛМП вырабатывает локальную меру подобия $q_{i,j}$, которая поступает на вычислитель интегральных мер подобия (ВИМП), реализующий описанный ДП-алгоритм, при этом начальные значения интегральных мер $\mathcal{F}_{0j}^k$ и $\mathcal{F}_{i0}^k$ берутся равными нулю. Для каждого i -го отсчета реализации ВИМП проводит расчет интегральной меры для всех отсчетов $0 \leq j \leq N$ каждого k -го эталона. После окончания речевой команды ($i=M$) ВИМП рассчитывает последний столбец $\{\mathcal{F}_{Mj}^k\}$, значения которых с индексом $j=N$ переписываются в ЗУ конечных интегральных мер подобия (КИМП). Решающее устройство (РУ) определяет $\max_k \{\mathcal{F}_{M,N}^k\}$ и по-

сле сравнения его с некоторым порогом ε принимает решение о номере k произнесенной речевой команды.

В настоящее время существуют различные модификации ДП-метода временной нормализации, которые используют различные ограничения (функции деформации оси времени, весовых функций параметров, отражающих значимость отдельных сегментов) и другие фонетические знания. Все модификации метода направлены на повышение точности распознавания речи, уменьшение времени принятия решений и упрощение технической реализации ДП-процессора. (Подробнее изложение разнообразных аспектов приложения ДП-метода к задаче распознавания и смысловой интерпретации речи см. в [Винцюк, 1987].)

5.4. Формантный синтезатор речевых сигналов

Формантный метод синтеза речевых сигналов обладает рядом важных достоинств:

высокой степенью адекватности естественным процессам звукообразования в речевом тракте, что потенциально позволяет получить синтетическую речь, не отличимую от естественной;

сравнительной простотой и высокой степенью изученности аналитических связей между форманными и формантным представлениями процессов речеобразования, что позволяет разработать относительно простые и высокоэффективные алгоритмы синтеза речи по тексту;

наибольшей степенью компрессии информационного потока (минимальный информационный поток, необходимый для восстановления качественной речи по формантным данным, 600 бит/с по сравнению с 1200 бит/с для методов линейного предсказания);

сравнительной простотой аналоговой и цифровой реализации формантного синтезатора (например, для реализации цифрового формантного синтезатора достаточно только одной БИС, а синтезатора на основе линейного предсказания — не менее двух).

Формантный синтезатор включает два основных функциональных блока моделирования источников возбуждения и передаточной функции речевого тракта. В зависимости от того, насколько полно будут промоделированы характеристики источников возбуждения и передаточной функции, зависит качество синтезированной речи.

Качество синтезированной речи зависит от правильного выбора формы импульсов голосового возбуждения формантных фильтров. Форма импульсов голосового возбуждения должна соответствовать форме естественных сигналов возбуждения, вырабатываемых голосовыми связками. Этому условию во многом удовлетворяет модель источника возбуждения, работа которого основана на нейроронаксической теории голосообразования [Лобанов, 1988]. Согласно предложенной модели с началом каждого периода на голосовые связки одновременно действует сила нервного возбуждения $F_{нв}$ (рис. 5.10, а), раздвигающая голосовые связки. Возвращение голосовых связок в сомкнутое состояние происходит под действием сил Бернулли F_B (рис. 5.10, б). Траектория движения краев голосовых связок S_0 подчиняется квадратичному закону и описывается параболической кривой (рис. 5.10, в).

Описанная процедура генерирования импульсов голосового возбуждения моделируется схемой, приведенной на рис. 5.11. Управляемый генератор импульсов (УГИ) голосового возбуждения вырабатывает прямоугольные импульсы, период следования, асимметрия и скважность которых задаются сигналами управления Упр. T_0 , Упр. α и Упр. T_1 . Срезом импульса производится установочный триггер T . Полученный на выходе сумматора сигнал, имитирующий суммарное действие сил $F_{нв}$ и F_B , дважды интегрируется (Инт1, Инт2); в результате моделируются изменения скорости и перемещения краев голосовых связок. При достижении выходным сигналом нулевого уровня индикатор нуля (И «0») осуществляет сброс интегратора Инт1 и триггера, а схема возвращается в исходное состояние.

Схема позволяет осуществлять независимое управление периодом, скважностью и асимметрией импульсов голосового возбуждения путем задания значений T_0 , T_1 , α . Ее использование в формантном синтезаторе обеспечивает получение высококачественного звучания и имитацию различных тембров голоса.

Другим весьма важным функциональным звеном синтезатора, от которого зависит как качество синтезированной речи, так и сложность технической реализации синтезатора, является формантный фильтр. В соответствии с акусти-

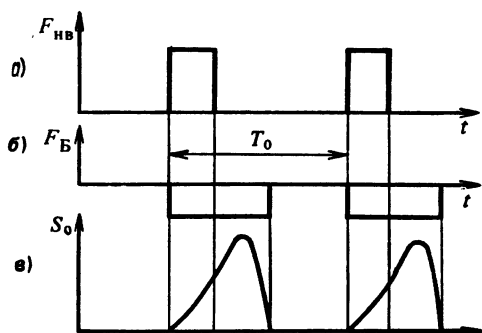


Рис. 5.10. Модель формирования импульсов голосового возбуждения

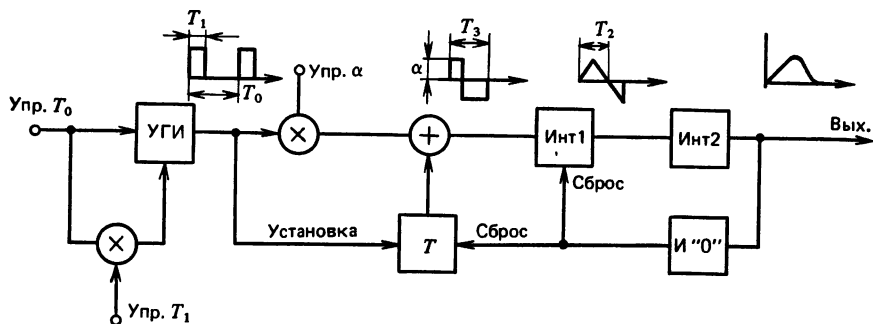


Рис. 5.11. Схема источника импульсов голосового возбуждения

ческой теорией речеобразования [Фант, 1964] в каждой ветви синтезатора рекомендуется использовать формантные фильтры, представляющие собой резонансные звенья с передаточной функцией

$$H_n(p) = (\tau_n^2 p^2 + 2\varepsilon_n \tau_n p + 1)^{-1},$$

где коэффициенты $\tau_n = 1/2\pi F_n$ и $\varepsilon = \pi B_n \tau_n$ определяют частоту F_n и полосу пропускания B_n n -го формантного фильтра (см. § 4.3). Это выражение совпадает с передаточной функцией активного RC -фильтра второго порядка, электрическая схема которого приведена на рис. 5.12. Для такого фильтра при условии, что коэффициент передачи напряжения $K_U = 3$, задаваемый соотношением сопротивлений R_1 , R_2 и $R' \gg R$, передаточная функция имеет вид

$$H(p) = 3 \left(C^2 R^2 p^2 + \frac{2CR^2}{R'} p + \frac{R^2}{R'^2} + 1 \right)^{-1},$$

а частота настройки F_n и полоса пропускания B_n фильтра определяются по формулам

$$F_n = \frac{1}{2\pi RC}; \quad B_n = \frac{1}{\pi R' C}.$$

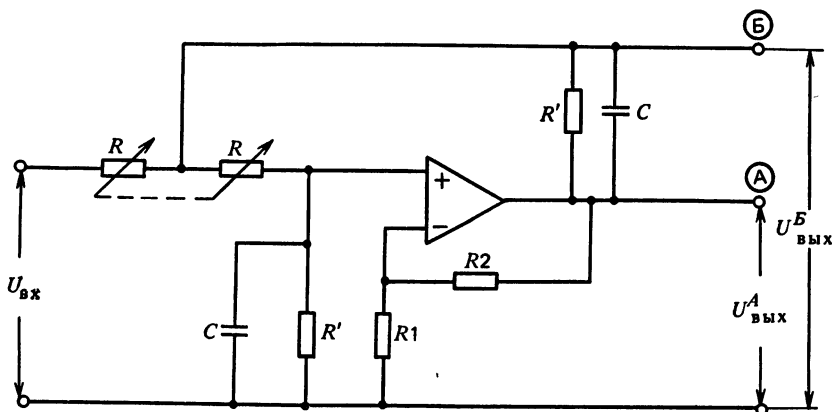


Рис. 5.12. Схема активного RC -фильтра второго порядка

Таблица 5.2.

Параметры формантных фильтров

Частоты формант, кГц	Ротовые				Фрикативные		Носовые	
	$\Phi_p 1$	$\Phi_p 2$	$\Phi_p 3$	$\Phi_p 4$	$\Phi_f 1$	$\Phi_f 2$	$\Phi_n 1$	$\Phi_n 2$
$F_n^{\min}$	0,2	0,6	1,7	2,5	1,8	2,2	0,2	0,6
$F_n^{\max}$	0,8	2,2	3,5	4,5	4,0	6,0	0,3	2,2
B_n	0,08	0,1	0,15	0,2	0,4	0,6	0,06	0,3

В соответствии с формантной моделью, приведенной на рис. 4.3, в схеме формантного синтезатора используется три параллельных канала, состоящих из последовательно соединенных формантных фильтров:

канал ротовых формант — $\Phi_p 1, \Phi_p 2, \Phi_p 3, \Phi_p 4$;

канал фрикативных формант — $\Phi_f 1, \Phi_f 2$;

канал носовых формант — $\Phi_n 1, \Phi_n 2$.

Параметры F_n и B_n для этих формантных фильтров приведены в табл. 5.2 (мужской голос).

Описанная реализация формантного фильтра в виде аналогового модифицированного активного RC-фильтра второго порядка широко использовалась при построении лабораторных и промышленных образцов формантных синтезаторов, обеспечивая высокое качество синтетической речи при сравнительно простоте технической реализации.

Общая тенденция микроминиатюризации электронных приборов и стремление к максимальному повышению технологичности изготовления и настройки аппаратуры стимулировали разработку цифровых методов синтеза речевых сигналов. При цифровой реализации, как и при аналоговой, основную трудность представляет моделирование характеристик формантного фильтра. Проведенные исследования показали, что по своим характеристикам наиболее близкими к аналоговым формантным фильтрам являются цифровые фильтры второго порядка, описываемые уравнением

$$y(n) = [2y(n-1) - x(n-1)]K_1 - y(n-2)K_2 + x(n),$$

где $x(n)$, $x(n-1)$ — входные сигналы фильтра; $y(n)$, $y(n-1)$, $y(n-2)$ — выходные сигналы; n , $n-1$, $n-2$ — текущий и предшествующие отсчеты дискретного времени; K_1 , K_2 — коэффициенты, определяющие частоту настройки и полосу пропускания фильтра. Уравнению соответствует схема цифрового фильтра на рис. 5.13.

Коэффициенты фильтра K_1 , K_2 связаны с частотой F_n и полосой пропускания B_n n -й форманты следующими соотношениями:

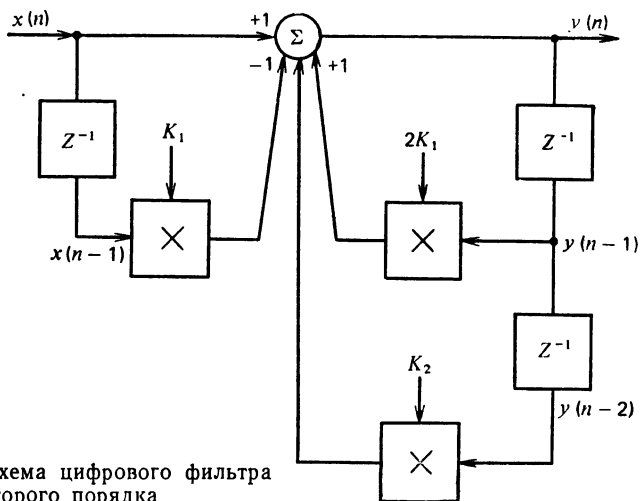
$$K_1 = (1 - \pi B_n / f_T) \cos 2\pi F_n / f_T,$$

$$K_2 = (1 - \pi B_n / f_T)^2,$$

где f_T — частота дискретизации сигнала. Эти формулы используются для расчета коэффициентов K_1 , K_2 при перестройке формантных фильтров в процессе синтеза речи.

Схема формантного синтезатора, построенная на основе цифровых фильтров (ЦФ), приведена на рис. 5.14. Широкий диапазон регулировки как резонансной частоты, так и полосы пропускания ЦФ позволяет использовать одни и те же ЦФ при построении каналов ротовых, носовых и фрикативных формант.

При синтезе гласных, носовых, аспиративных звуков фильтры ЦФ1—ЦФ4 настраиваются соответственно на резонансные частоты первой, второй, третьей

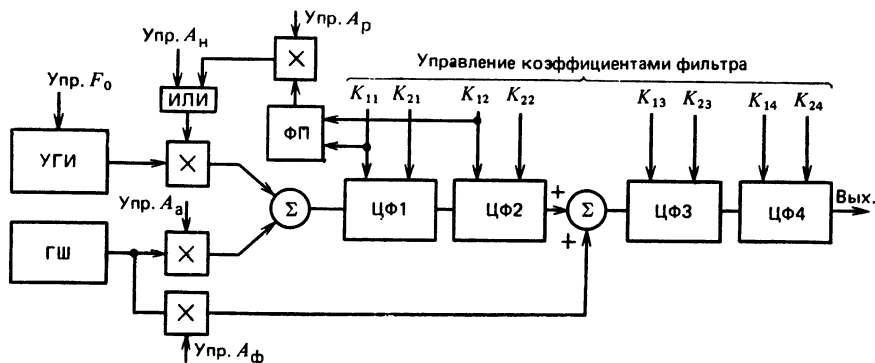


и четвертой ротовых или носовых формант. При синтезе фрикативных звуков настраиваются фильтры ЦФЗ и ЦФ4 на частоты первой и второй фрикативных формант. Изменение резонансных частот и полос пропускания ЦФ производится путем изменения коэффициентов K_{1i} и K_{2i} соответствующих фильтров ($i=1-4$).

При образовании гласных или носовых звуков на вход ЦФ через сумматор подаются импульсы голосового возбуждения с выхода управляемого генератора импульсов (УГИ) голосового возбуждения. Частота следования импульсов изменяется в соответствии с сигналом Упр. F_0 , а их амплитуда — в соответствии с сигналом Упр. A_p и Упр. A_n .

Для выравнивания общего коэффициента передачи цепочки ЦФ при синтезе различных звуков сигнал Упр. A_p дополнительно умножается на сигнал, вырабатываемый от функционального преобразователя (ФП), уровень которого зависит от значений коэффициентов K_{11} и K_{21} .

Для синтеза аспиративного шума на вход ЦФ через сумматор подается сигнал генератора шумового возбуждения (ГШ). Амплитуда шумового возбуж-



дения моделируется сигналом Упр. A_a . Для синтеза фрикативного шума сигнал ГШ через сумматор подается на вход ЦФЗ. Управление амплитудой фрикативного шума производится сигналом Упр. A_f .

Схема на рис. 5.14 реализуется с помощью специализированного вычислительного устройства, построенного на одной БИС.

5.5. Универсальный фонемный синтезатор речевых сообщений

Фонемные синтезаторы речевых сообщений (синтезаторы речи по тексту) являются универсальным средством человека-машинного речевого диалога в смысле возможности синтеза речи неограниченного словарного состава непосредственно по орфографическому тексту любого содержания.

Алгоритм работы фонемного синтезатора следующий. Орфографический текст, поступающий от ЭВМ, подвергается разметке — выделению в тексте синтагм. Каждой выделенной синтагме присваивается определенный интонационный тип. Производится «склеивание» предлогов, союзов и частиц со знаменательными словами. Размеченный таким образом орфографический текст поступает на преобразователь орфографический текст — фонемный. В этом блоке по определенным правилам производятся подстановки «сочетание букв или цифр — сочетание фонем». В результате работы преобразователя получаем фонемный текст сообщений.

Для озвучивания фонемного текста требуется рассчитать набор формантных и просодических характеристик. Это осуществляется следующим образом (см. § 4.2):

- производится разметка фонемного текста — в каждой синтагме выделяются акцентные группы;

- по интонационному типу синтагмы определяются соответствующие мелодический, ритмический и энергетический контуры акцентных групп (хранятся в библиотеке портретов/интоном),

- выделяются слоги, рассчитывается ритмический контур и длительности всех фонем слога, строятся его мелодический и энергетический контуры;

- рассчитываются и модифицируются (осуществляется привязка их к конкретной структуре слога) формантные параметры фонем на базе информации, содержащейся в портретах фонем;

- рассчитываются значения характеристических точек траектории движения параметров внутри слога;

- по характеристическим точкам строятся непрерывные траектории движения параметров и передаются в формантный синтезатор речевых сигналов.

В основе метода синтеза формантных параметров лежит представление фонемы в виде динамического формантного портрета. *Формантный портрет j -фонемы* (рис. 5.15) задается набором значений 10 формантных параметров на пяти последовательных временных сегментах: T_0 — начальный; T_1 — основной; T_2 , T_3 — дополнительные и T_4 — конечный. В фонемном портрете $T_0 = T_4 = 0$, $T_1 > 0$ всегда, а $T_2, T_3 \geq 0$ в зависимости от типа фонемы.

На основном и дополнительном сегментах формантные портреты задаются тремя параметрами: собственным значением P_i^{jk} k -го параметра на i -м сегменте j -й фонемы; коэффициентом коартикуляции α_i^{jk} ; длительностью переходного процесса τ_i^{jk} ; общей для всех параметров собственной длительностью сегмента T_i^j . На начальном сегменте задаются значения P_0^{jk} , α_0^{jk} , τ_0^{jk} , а на конечном — только значение τ_4^{jk} .

Для автоматического синтеза формантный портрет представляется в виде таблицы, в которой для каждого фонемного сегмента (строка таблицы) указываются его длительность и значения указанных выше параметров каждого формантного портрета (столбец таблицы). Для синтеза русской речи требуется 42 таблицы размерностью 23×5 .

Формантные портреты являются универсальной контекстно-независимой формой задания акустических характеристик фонем при синтезе речи. Для

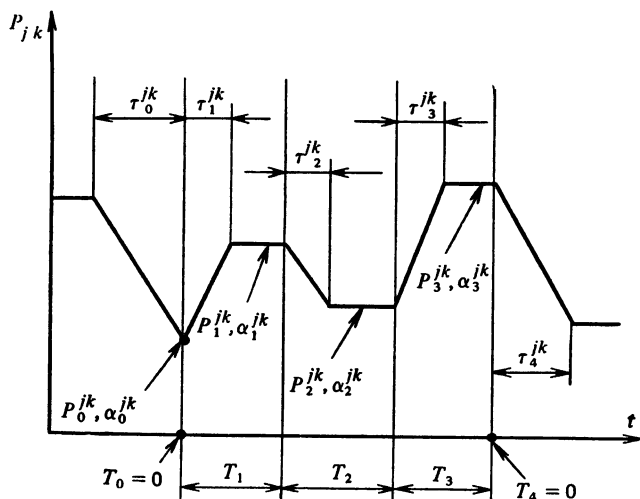


Рис. 5.15. Формантный портрет фонемы

учета контекстуальной изменчивости фонем разработан комплекс правил модификации формантных портретов фонем в речевом потоке.

Существует два основных вида изменчивости фонем в речевом потоке: комбинаторная и позиционная. Основными причинами, вызывающими комбинаторные изменения формантных параметров, являются эффекты коартикуляции и редукции. В основе этих эффектов лежит принцип параллельности работы каналов управления артикуляцией в процессе образования гласных и согласных фонем слога. На акустическом уровне взаимодействие этих каналов проявляется в закономерных изменениях формантных частот каждой фонемы в зависимости от ее окружения и темпа речи. Наряду с общими чертами механизмы коартикуляции и редукции имеют существенные различия. Коартикуляционным изменениям подвергаются в большей степени согласные фонемы, в меньшей — гласные. Эффект коартикуляционных изменений проявляется в основном независимо от темпа речи. Эффект редукционных изменений гласных и согласных проявляется только при достаточно быстром темпе в силу инерционности движения артикуляторов. На акустическом уровне редукция кроме уменьшения длительности фонем проявляется в изменениях формантных частот гласных под действием окружающих согласных.

Расчет формантных частот согласных F_n^c с учетом их коартикуляционной изменчивости внутри открытого слова $C_n \dots C_2 C_1 C_0$ осуществляется по следующей рекуррентной формуле

$$F_n^c = \alpha_n^c F_{n-1} + (1 - \alpha_n^c) P_n^c.$$

Здесь n — порядковый номер согласной фонемы в слоге начиная от гласной с индексом 0; α_n^c и P_n^c — коэффициент коартикуляции и собственная частота согласной фонемы, значения их берутся из таблиц фонемных портретов.

Для расчета коартикуляционных значений формантных частот гласных используется формула

$$F^r = \alpha^r (\gamma_1 P^{c1} + \gamma_2 P^{c2}) + (1 - \alpha^r) P^r,$$

где P^r , α^r — собственная частота и коэффициент коартикуляции гласной фонемы; P^{c1} , P^{c2} — собственные частоты согласных, стоящих соответственно слева и справа от гласной; γ_1 , γ_2 — весовые множители. Значения P^r , α^r , P^c берутся из таблиц фонемных портретов.

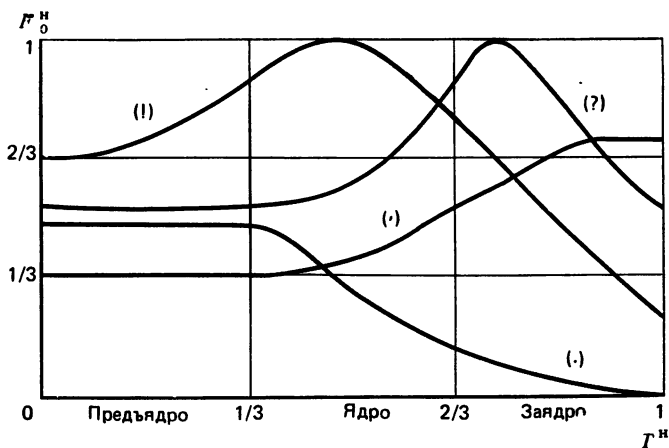


Рис. 5.16. Методические контуры четырех интонационных типов конечной акцентной группы

Использование описанных закономерностей при синтезе речи позволяет алгоритмизировать процесс порождения комбинаторных аллофонов, вызываемых коартикуляционными и редуцированными эффектами, и осуществлять расчет связанных с ними изменений формантных частот гласных и согласных фонем. Остается, однако, не столь значительное, но все же заметное число аллофонов, обусловленных позиционной изменчивостью амплитудно-временных характеристик фонем. Для алгоритмизации процесса порождения позиционных аллофонов разработаны частные правила, с помощью которых моделируются при определенных условиях потеря гласной вставки у сонорных и взрывных согласных, ослабление и глушение сонорных, потеря взрыва и удвоение фонем, а также некоторые другие явления, характерные для слитной речи.

Описанные особенности структуры формантных портретов фонем и правила их модификации в речевом потоке послужили основой для разработки алгоритмов синтеза формантных параметров по фонемному тексту.

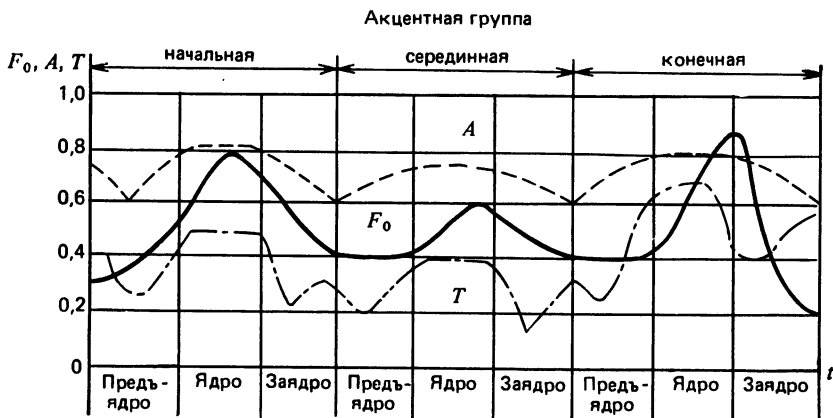


Рис. 5.17. Портрет интонеми общего вопроса

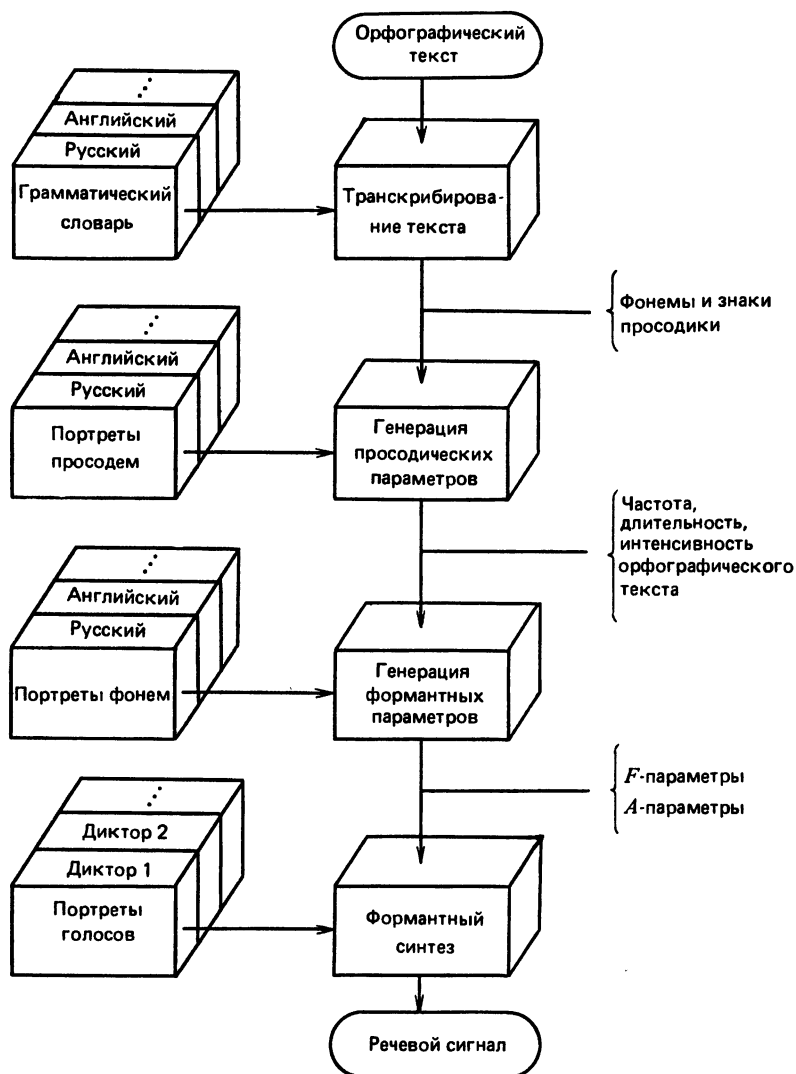


Рис. 5.18. Общая схема синтеза многоязычной речи

Рассмотрим особенности алгоритма синтеза просодических характеристик (мелодика, ритмика и энергетика) речи по тексту. Просодические характеристики описываются текущими значениями частоты основного тона $F_0(t)$, длительностью $T(t)$ и интенсивностью $A(t)$ фонемных сегментов (см. § 4.2). Просодические характеристики речи являются средством для реализации надфонемных (суперсегментных) языковых явлений, таких как ударение и интонация.

Минимальным элементом речи, для которой определено понятие интонационного типа, является синтагма. Совокупность просодических характеристик,

с помощью которых описывается интонационный тип синтагмы, названа *портретом интономы*. Портреты интоном задаются набором контуров мелодики, ритмики и энергетики, которые складываются из нормированных контуров входящих в нее акцентных групп: начальной, конечной и срединной. В свою очередь, контур каждой акцентной группы складывается из трех последовательных сегментов: предъядра, ядра и заядра. Мелодические контуры для одной акцентной группы приведены на рис. 5.16, а на рис. 5.17 — портрет для интонации общего вопроса.

Для удовлетворительного описания просодических характеристик широкого класса устных текстов достаточно задать портреты 16 интоном, являющиеся подтипами основных интонационных типов (см. § 4.2).

Просодический портрет описывает интонацию синтагмы независимо от числа входящих в нее акцентных групп. Если их число больше трех, то все акцентные группы между начальной и конечной являются срединными. Если акцентных групп две, то первая из них начальная, а вторая — конечная. Если акцентная группа одна, то она конечная. Сама акцентная группа описывается одним и тем же контуром независимо от числа слогов, входящих в нее. Если число слогов в предъядре или заядре больше трех, то добавляются срединные слоги. При меньшем числе из рассмотрения исключаются вначале срединный слог, затем начальный и конечный. Если предъядро или заядро отсутствуют, то их функцию берет соответственно левый или правый сегмент ядра.

Следовательно, алгоритм синтеза просодических характеристик речи по тексту может быть представлен как композиция алгоритмов разметки фонемного текста на интонационно-значимые сегменты и алгоритмов синтеза ритмического и энергетического контуров фразы на основе табличных портретов интоном.

Описанный алгоритм синтеза речи по тексту универсален также в смысле возможности синтеза многоязычной речи (рис. 5.18). Переход от одного языка к другому не связан с перестройками общей структуры алгоритма синтеза речи. При переходе к другому языку целиком или частично производится замена библиотеки портретов фонем и интоном, а также программы преобразования орфографического текста в фонемный.

На первом этапе орфографический текст преобразуется по определенным правилам в размеченный знаками просодики фонемный текст. При этом используются грамматические и лексические знания, заложенные в словаре. На втором этапе на основе портретов просодом и правил их трансформации в данном контексте генерируются просодические характеристики. На третьем формируются формантные параметры с использованием портретов фонем и правил их модификации под действием коартикуляции, редукции и ассимиляции. Полученные таким образом параметры поступают на формантный синтезатор речевых сигналов, параметры которого могут перестраиваться для задания желаемой индивидуальности синтаксического голоса.

Глава 6.

Состояние разработки и производства систем речевого общения

Б. М. Лобанов

6.1. Обзор промышленных систем речевого общения

Начало промышленного выпуска СРО в разных странах относится к середине 70-х — началу 80-х годов. К настоящему времени число различных промышленных СРО достигло нескольких десятков и продолжает расти. Одновременно продолжается рост сбыта СРО.

В США и Японии выпускаются различные СРО: стоимостью от 99 до 99 тыс. дол., имеющие габаритные размеры от одной БИС до стойки оборудо-

Название устройства (фирма, страна)	Метод синтеза речи	Объем словаря	Входная информация	
			вид	объем 1 с речи
DEC-talk (Dec., США)	Фонемно-формантный	Неограниченный	Орфографический текст	80
KPM (KURZ WEIL, США)	То же	То же	То же	80
M-140 (SPECH. TECHN. CORP, США)	"	120 слов	"	80
PROSE-2000 (TSS, США)	ЛПК, фонемный	Неограниченный	Размеченный текст	80
SPARTE (CNET, Франция)	ЛПК, дифонный	То же	То же	80
SPEAK-N-SPELL (TEXAS INSTR, США)	ЛПК, компилятивный	200 слов	Номера слов и фраз	1200
TYPE-N-TALK (VOTRAX, США)	Фонемно-формантный	Неограниченный	Размеченный текст	80 (слоговая)
EKO-11 (STREET ELECTR, США)	ЛПК, фонемный	То же	То же	80
Фонемофон-4Т (СССР)	То же	"	Орфографический текст	80
15KC2000-14 (СССР)	"	"	То же	80
MCP-1 (СССР)	ЛПК; компилятивный	100 слов	Номера слов или фраз	1300
Фонемофон-5 (СССР)	То же	То же	То же	80
ФС-05 (СССР)	"	"	"	80
PETER-C (СССР)	Формантный, компилятивный	Неограниченный	Размеченный текст	2400

вания, потребляемую мощность от долей ватта до 1 кВт, назначение — от игрушек до телефонных справочных систем. Соответственно меняются и качественные показатели: синтез речи — от машиноподобного воспроизведения нескольких десятков слов до синтезаторов речи неограниченного словаря с управляемыми характеристиками голоса, анализ речи — от распознавания 10 команд с надежностью 75—85 % до систем понимания речи со словарем в 1000 слов и надежностью 99 %. В США есть небольшие фирмы, которые занимаются исключительно системами распознавания и синтеза речи (VOTRAX, SRS, VOTAN, KAI), а также крупные компьютерные фирмы (TTI, DEC, Intel, IBM), имею-

Разборчивость речи, %	Синтезируемые		Телефонный интерфейс	Сфера применения	Источник
	языки	голоса			
92 (слоговая)	Английский	3 мужских, женский, детский	Есть	АСУ, САПР, АРМ	Dagrow, 1984
—	"	Мужской	Нет	Читающая машина для слепых	Кейтер. 1985
—	"	"	"	ПЭВМ, АРМ	"
63 (слоговая)	"	"	"	ПЭВМ	"
—	Французский	Мужской, женский	"	ПЭВМ	"
—	Английский	Мужской	"	Обучение	"
27 (слоговая)	"	"	"	ПЭВМ	"
—	"	"	"	"	"
94 (звуковая)	Русский	Мужской	Есть	АСУ, САПР	Лобанов, 1988
90 (звуковая)	"	"	"	То же	Лобанов, 1982
95 (словесная)	Русский	"	"	Технологическое оборудование	Смирнов и др., 1984
96 (звуковая)	Русский, английский	3 мужских, 2 женских	Нет	САПР, АРМ, ПЭВМ	Лобанов и др., 1986
93 (слоговая)	"	"	"	"	"
—	Русский	Мужской	"	Речевой вывод ЭВМ	То же
93 (звуковая)	"	"	"	ПЭВМ	Богино и др., 1986

шие собственные подразделения по разработке СРО. Большое внимание уделяется разработке средств распознавания и синтеза речи в Японии (фирмы NEC, Matsuchiba, San'yo). В Западной Европе для проведения работ по созданию СРО создан специальный консорциум в составе 10 фирм (Plessay, Ferranty, GEC Marconi, Thorr-EMI, Pocal, Olivetti и др.).

По прогнозу фирмы International Resource Development (США) объем продажи устройств речевого ввода-вывода в США резко возрастет в начале 90-х годов, к 1992 г. составит 4 млрд. дол. [Dagrow, 1984]. Ожидается, что к этому времени около 60 % всех продаваемых вычислительных систем будут

Название устройства (фирма, страна)	Метод анализа и распознавания	Вид распозна- ваемой речи	Объем словаря, слова	Надеж- ность рас- познава- ния, %
CSE-1060 (CGK, ФРГ)	Полосный, ДП	Изолированные слова	250	99
DP-200 (NEC, Япония)	Полосный	То же	500	98,8
KVS-3000 (Kurzweil, AI, США)	Фонетические характеристи- ки, СММ	Изолированные слова и слит- ная речь	1000	95
SR-150 (NEC, Япония)	—	То же	149	99 (цифры)
SR-2000 (NEC, Япония)	—	Изолированные слова	19	99 (цифры)
VRT 101/3 (Interstate, США)	Полосный	То же	89	97
THRESHOLD-600 (Threshold, США)	Наклоны спек- тра, ДП	„	250	98,6
VOTAN-1000 (Votan, США)	БПФ, ДП	„	300	99
VERBEX-1800 (Verbex, США)	Корреляцион- ная функция, ДП	Изолированные слова и слит- ная речь	39	99,8
VERBEX-3000 (Verbex, США)	Фонетические характеристи- ки, СММ	То же	1000	99
ПС-7801 (СССР)	Спектр Уолша, локальное ДП	Изолированные слова	250	98 на 100 слов
ДИС-87 (СССР)	Полосный, гра- диентный спуск	То же	200	96
ИКАР (СССР)	Производная спектра, ДП	„	200	97
МАРС-М (СССР)	Формантный, ДП	Изолированные слова (слитная речь)	100 (50)	99,3 (96)
МАРС-1 (СССР)	Формантный, ДП	Изолированные слова	200	97
МАРС-3 (СССР)	Полосный, гра- диентный спуск	То же	250	97,5 (мик- рофон) 95 (теле- фон) на 100 слов
УРРК (СССР)	Полосный, ДП	„	250	95

познавания речи

Допустимый уровень помех, дБ	Способ адаптации к директору	Наличие телефонного интерфейса	Наличие синтезатора речи	Область применения	Источник
70	—	Нет	Нет	Производство	Трунин-Донской, 1984
85	(1—4)-кратное произнесение	"	"	То же	То же
70	3-кратное произнесение	"	"	"	Kurzweil, 1985
90	Однократное произнесение	Есть	Есть	Терминал ЭВМ	Трунин-Донской, 1984
80	Произвольный диктор	"	"	Запросы по телефону	То же
90	5-кратное произнесение	"	Нет	Производство торговля	"
85	10-кратное произнесение	Нет	"	Производство	"
100	2-кратное произнесение	Есть	Нет	Производство, запросы по телефону	Winn, 1987
80	Произвольный диктор	"	"	Запросы по телефону	То же
70	3-кратное произнесение	Нет	"	Производство	Ross et al., 1984
70	Однократное произнесение	"	"	Вычислительные комплексы	Трунин-Донской 1987
60	То же	"	"	ПЭВМ	То же
80	"	"	"	АРМ	Петров и др., 1980
100	"	"	Есть	САПР, АРМ	Дегтярев и др. 1987
70	"	Нет	Есть	САПР	Афанасьев и др., 1984
70	"	Есть	"	АСУ, АРМ	Трунин-Донской, 1987
70	"	Нет	"	Специальное	То же

Название устройства (фирма, страна)	Метод анализа и распознавания	Вид распозна- ваемой речи	Объем словаря, слова	Надеж- ность рас- познава- ния, %
РЕЧЬ-1 (СССР)	Знак производ- ной спектра, ДП	Изолированные слова	200	95
РЕЧЬ-121 (СССР)	То же	Изолированные слова (слитная речь)	600	99 (93 на 200 слов)
БАРС (СССР)	Полосный, ДП	Изолированные слова	300	97
СИБИРЬ-1 (СССР)	Полосный, ДП	То же	300	95
СИРИУС-1 (СССР)	Клиппирован- ный сигнал, ДП	„	100	92

оснащены устройствами речевого ввода-вывода, причем значительная доля рынка придется на системы обработки текста.

Как уже отмечалось, СРО строятся на базе специализированных устройств: синтезаторов и анализаторов (распознавателей) речи. Поэтому в дальнейшем, характеризуя особенности устройств, мы будем пользоваться отдельными наборами параметров для синтезаторов и анализаторов речи.

Характеристики отечественных и зарубежных синтезаторов речи приведены в табл. 6.1. Таблица охватывает практически все отечественные и наиболее типичные зарубежные разработки синтезаторов речи на конец 1988 г.

Синтез речевых сигналов в большинстве устройств осуществляется методом линейного предиктивного кодирования (ЛПК) либо формантным; при этом каждый из них используется как при синтезе ограниченного словаря — компилятивный метод, так и при универсальном синтезе — фонемный или дифонный метод.

Объем словаря у компилятивных синтезаторов от 100 до 200 слов, у универсальных — неограниченный.

Входной информацией для компилятивных синтезаторов являются номера слов или фраз из заранее заданного словаря, а для универсальных — это обычный орфографический текст сообщения, как правило, предварительно размеченный дополнительными знаками ударения и интонации. Важным параметром синтезаторов является объем входной информации, необходимый для синтеза 1 с речи. Этот параметр определяет емкость памяти, необходимую для хранения слов заданного словаря (примерно 2 слова на 1 с речи). Для компилятивных синтезаторов объем информации от 1200 до 3200 бит/с, для универсальных — 80 бит/с.

Разборчивость синтезированной речи характеризуется процентом правильно воспринятых аудиторами звуков, слогов, слов или фраз на специально подобранных текстах. В настоящее время нет единой методики определения разборчивости синтезированной речи, поэтому данные, приведенные в табл. 6.1, в значительной степени неоднородны и субъективны. Известна только одна попытка применения единой методики для измерения слоговой разборчивости речи трех различных синтезаторов TYPE-N-TALK, PROSE-2000 и DEC-talk [Пизони и др., 1985]. По этой же методике проведено измерение слоговой разборчивости отечественных синтезаторов ФОНЕМОФОН-П и ФОНЕМОФОН-5.

Допустимый уровень помех, дБ	Способ адаптации к директору	Наличие телефонного интерфейса	Наличие синтезатора речи	Область применения	Источник
75	Однократное произнесение	Нет	Есть	САПР, АСУ, АРМ	Винцюк и др., 1982
90	(1—10)-кратное произнесение	„	„	САПР, АСУ, АСУТП	Винцюк, 1987
70	Однократное произнесение	„	„	САПР	Трунин-Донской, 1987
75	То же	Нет	„	Диспетчерские системы	То же
75	„	„	„	Специальное	„

Важными параметрами с точки зрения потребителя являются возможность синтеза речи на различных языках с различными типами голосов, в том числе синтеза женского голоса. Сфера применения синтезаторов резко расширяется, если дополнить его телефонным интерфейсом, позволяющим абоненту получать информацию по телефонной коммутируемой сети.

Характеристики зарубежных и отечественных устройств распознавания речи приведены в табл. 6.2. Таблица охватывает практически все отечественные и наиболее типичные зарубежные разработки на конец 1988 г.

Анализ речевых сигналов в большинстве устройств осуществляется с помощью спектрально-полосных анализаторов различных модификаций, а анализ сообщений (распознавание слов) — с помощью ДП-анализаторов. В некоторых устройствах анализ речевых сигналов осуществляется путем измерения корреляционных функций, формантных параметров, плотности нулей клиппированного сигнала. В последнее время анализируются также различные фонетические характеристики речи: звонкость, шумность, гласность, взрывность и др. При распознавании кроме ДП-метода используются также методы градиентного спуска и скрытых марковских процессов (СМП). Большинство существующих устройств рассчитано на распознавание изолированно произнесенных слов (речевых команд), причем средний объем словаря порядка 200 слов и только для некоторых созданных в последние годы устройств достигается возможность распознавания слитной речи (или квазислитной с микропаузами) и увеличивается максимальный объем словаря до 1000 слов.

Под надежностью распознавания подразумевается процент правильно распознанных слов из заданного словаря в заданных условиях работы. В настоящее время нет единой методики тестирования устройств по надежности распознавания. Поэтому данные, приведенные в табл. 6.2, в значительной степени субъективны: неизвестны степень обученности дикторов, данные о микрофоне и т. д. Во многом субъективны и приведенные в таблице данные о допустимом уровне стационарных помех.

Наличие телефонного интерфейса в устройстве предполагает не только введение специальных блоков сопряжения со стандартной телефонной сетью, но и принятия специальных мер для эффективной обработки телефонных сигналов. Наличие синтезатора речи расширяет сферу применения устройства распознавания, позволяя реализовывать СРО в полном объеме.

6.2. Отечественные системы речевого общения

Первые опытные образцы промышленных СРО в СССР выпущены в начале 80-х годов. К ним относятся речевые терминалы серий ИКАР [Петров и др., 1980] и МАРС [Афанасьев и др., 1984], синтезаторы речи серии ФОНЕМОФОН [Лобанов, 1982] и система речевого диалога серии РЕЧЬ [Винцюк и др., 1982]. Во второй половине 80-х годов выпуск промышленных СРО серий ФОНЕМОФОН, МАРС и РЕЧЬ достиг уже нескольких сотен.

6.2.1. Синтезаторы речи серии ФОНЕМОФОН. Принципы работы синтезаторов этой серии изложены в § 5.5.

Синтезаторы речи ФОНЕМОФОН-4Т — качественно новая ступень в развитии устройств речевого вывода. Они реализованы на микроЭВМ или микропроцессорных наборах. Это позволило постоянно совершенствовать и уточнять алгоритмы синтеза речи, не изменяя конструкции и схемы устройств, а также постоянно уменьшать габаритные размеры, массу и потребляемую мощность по мере совершенствования БИС, входящих в микропроцессорные наборы.

В устройстве ФОНЕМОФОН-4Т использован программно-аппаратурный подход к реализации синтеза речи и установления связи с пользователем. В состав устройства (рис. 6.1) в качестве функционального блока входит формантный синтезатор (ФС) речевых сигналов, являющийся функциональной моделью речевого тракта человека. Формирование сигналов управления синтезатором осуществляется программой преобразования «текст — формантные параметры», реализуемой процессором (ЦП) по тексту, считываемому с ОЗУ (2 Кбайт). Занесение текста в ОЗУ осуществляется через последовательный интерфейс (ПИ) с базовой ЭВМ системы. Программы синтеза речи и функционирования устройства в системе записаны в ПЗУ емкостью 4 Кслов.

В состав синтезатора речи включено телефонное автоматическое устройство (ТАУ), осуществляющее сопряжение синтезатора с телефонной сетью. ТАУ и ФС подключены к каналу через параллельный интерфейс И2. Таким образом, кроме основного назначения — речевого отображения поступающей информации — устройство с помощью ТАУ обеспечивает также взаимодействие ЭВМ с абонентами телефонной сети. При этом ТАУ выполняет роль своеобразного телефонного аппарата, с помощью которого ЭВМ может позвонить и передать информацию в речевом виде любому абоненту и, наоборот, абонент может позвонить ЭВМ для получения необходимой информации. Устройство может также работать непосредственно с оператором ЭВМ или системы. В этом случае

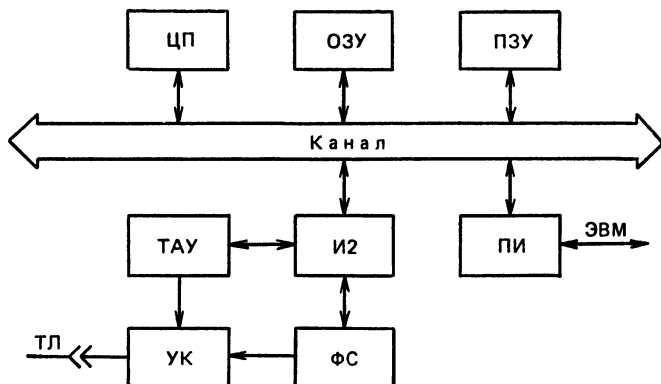


Рис. 6.1. Структурная схема речевого информатора ФОНЕМОФОН-4Т:

ЦП — центральный процессор; ОЗУ — оперативное запоминающее устройство; ПЗУ — постоянное запоминающее устройство; ПИ — последовательный интерфейс; И2 — параллельный интерфейс; ТАУ — телефонное автоматическое устройство; УК — устройство коммутации; ФС — формантный синтезатор; ТЛ — телефонная линия

ТАУ не используется, а к устройству подключается громкоговоритель либо головной телефон. Синтезатор ФОНЕМОФОН-4Т может применяться совместно с ЭВМ «Электроника 100-25» или СМ-4.

При подключении синтезатора речи ФОНЕМОФОН-4Т к телефонной сети он работает как речевой информатор. Установление с помощью ТАУ связи по телефонной линии и обмен информацией между речевым информатором и пользователем возможны по инициативе как ЭВМ — режим ЭВМ — абонент, так и абонента — режим абонент — ЭВМ.

В режиме ЭВМ — абонент ЭВМ передает в информатор номер телефона абонента. Информатор подключает к телефонной линии нагрузку (снимает трубку) и ждет сигнал «Готовность АТС». При поступлении сигнала «Готовность АТС» информатор «набирает» номер. После этого из телефонной линии могут прийти сигналы «Занято» или «Речь». По сигналу «Занято» информатор набирает другой номер или отключается от телефонной линии. Если поступил сигнал «Речь», это означает, что абонент снял телефонную трубку и ответил «Алло», «Слушаю» и т. д., при этом ЭВМ передает текстовое сообщение в информатор. Информатор преобразует текстовое сообщение в речевое и выводит его в телефонную линию. По окончании речевого сообщения режим ЭВМ — абонент может быть прерван по инициативе как ЭВМ, так и абонента.

Для прекращения диалога по инициативе ЭВМ сообщение, которое передается в информатор, сопровождается командой отключения информатора от телефонной линии. Если диалог прекращается по инициативе абонента, то в информатор из телефонной линии поступает сигнал «Отбой», который означает окончание диалога.

Наличие в информаторе (в блоке ТАУ) приемника тонального кода запроса позволяет абоненту запрашивать из ЭВМ необходимую ему информацию. Для этого абоненту следует набрать код тонального запроса.

При установлении телефонной связи по инициативе абонента абонент набирает номер телефона, присвоенный подключенному к телефонной линии информатору. Из телефонной линии в информатор поступает сигнал «Вызов». По этому сигналу информатор подключает к телефонной линии нагрузку (снимает трубку), ЭВМ передает кодовое текстовое сообщение в информатор. Информатор преобразует текстовое сообщение в речевое и выводит его абоненту. Дальнейшая работа производится аналогично описанному выше.

Решение ряда схемотехнических вопросов, направленных на достижение высокой технологичности изделий и повторяемости их характеристик, снижение массы и габаритных размеров позволило перейти на цифровые методы обработки сигналов. Так, к настоящему времени разработан синтезатор речи ФОНЕМОФОН-5 [Лобанов и др., 1986], выполненный на одноплатном модуле размером 135×210 мм, что позволило улучшить качественные характеристики синтетической речи по сравнению с моделью ФОНЕМОФОН-4Т.

6.2.2. Система речевого диалога РЕЧЬ-1. Система речевого диалога РЕЧЬ-1 предназначена для обеспечения двустороннего взаимодействия человека и ЭВМ посредством голоса, ориентирована на применение в АСУ и САПР, в составе АРМ конструктора и в других человеко-машинных системах сбора, обработки информации и управления.

Система речевого диалога РЕЧЬ-1 состоит из двух блоков микроЭВМ «Электроника-60М» вместе с периферийными устройствами и блоком анализа, синтеза и визуализации, в состав которого входит 16-разрядный спектральный анализатор, 8-байтовый формантный синтезатор речи, прибор ПИУ-2 на 16 знакомест для высвечивания результатов распознавания и служебной информации, микрофон, звуковая колонка, кнопка связи на гибком шланге, тумблер связи.

Питание системы осуществляется от стандартного блока питания БПС6-1, входящего в комплект микроЭВМ «Электроника-60М».

Система РЕЧЬ-1 структурирована и организована таким образом, что может быть использована и как универсальная микроЭВМ, и как диалоговая вычислительная система с речевым вводом и выводом информации. Основной

режим работы системы РЕЧЬ-1 — распознавание устных команд и синтез речи. Человек-оператор произносит в микрофон устные команды. Результат распознавания в форме текста высвечивается на индикаторной панели ПИУ-2 и(или) произносится синтезатором речи. Синтезатор речи также озвучивает различного рода служебную информацию, которая одновременно выдается на ПИУ-2.

Система РЕЧЬ-1 может работать в одном из следующих режимов:

- обучение распознаванию (О);
- собственно распознавание и синтез речи (Р);
- корректировка результатов обучения (К);
- проверка спектрального анализатора и качества предварительной обработки речи (Т);
- печать результатов обучения (Е),
- проверка синтезатора речи (С).

Переход из блока в блок осуществляется в режиме диалога пользователя с системой РЕЧЬ-1 посредством клавиатуры.

Распознавание речи в системе РЕЧЬ-1 основано на представлении 15-мс отрезков речевого сигнала в виде 16-разрядного двоичного кода, задающего знаки разности энергий в соседних спектральных каналах (знак производной спектра), и на вводе этих кодов в память микроЭВМ через каждые 15 мс.

В процессе ввода кодов в память микроЭВМ осуществляется оптимальная сегментация речевого сигнала с помощью ДП-метода, заключающаяся в таком разбиении последовательности кодов на подпоследовательности длиной не менее чем три кода, при котором суммарное отклонение кодов последовательности от средних кодов сегментов наименьшее. Последовательность из средних кодов оптимальных сегментов образует сокращенное описание реализации речевого сигнала, которое и используется непосредственно в режиме обучения и распознавания.

Эталоном слова объявляется сокращенное описание какого-либо произнесения этого слова. В процессе распознавания сокращенное описание предъявленного для распознавания сигнала сравнивается со всеми эталонами слов с помощью локального ДП. Ответом распознавания объявляется то слово, эталон которого дал наибольшее сходство с распознаваемым сигналом.

Другие модификации систем серии РЕЧЬ описаны в [Винько, 1987].

6.2.3. Речевой терминал МАРС-1. Устройство МАРС-1 предназначено для организации речевого диалога в человеко-машинных технологических системах. В устройстве МАРС-1 использован программно-аппаратурный подход к реализации распознавания речевых команд и синтеза речи.

В состав устройства (рис. 6.2) в качестве функционального блока входит анализатор речевых сигналов (АРС), предназначенный для формантного представления входного речевого сигнала. В основе работы АРС лежит метод ДП-сравнения реализации речевого сигнала с эталонами команд словаря в пространстве формантного описания и выбор ближайшего эталона. ДП-сравнение осуществляется процессором динамического программирования (ПДП). Программа принятия решения реализуется центральным процессором (ЦП), в качестве которого используется процессор М2 микроЭВМ «Электроника-60М». В этом же процессоре при синтезе речи осуществляется формирование сигналов управления формантным синтезатором (ФС) речевых сигналов по последовательности символов орфографического текста, считываемых с оперативно-запоминающего устройства (ОЗУ). Занесение текста в ОЗУ осуществляется через последовательный интерфейс (ПИ) либо с базовой ЭВМ системы, либо при автономном режиме работы МАРС-1 с алфавитно-цифровой клавиатуры (АЦК). В устройстве предусмотрена возможность визуального отображения информации на встроенный дисплей ИМГ-1-03. Программы функционирования, распознавания и синтеза речи записаны в ПЗУ емкостью 16 Кслов.

Основные узлы терминала МАРС-1 выполнены на микропроцессорном наборе серии 589.

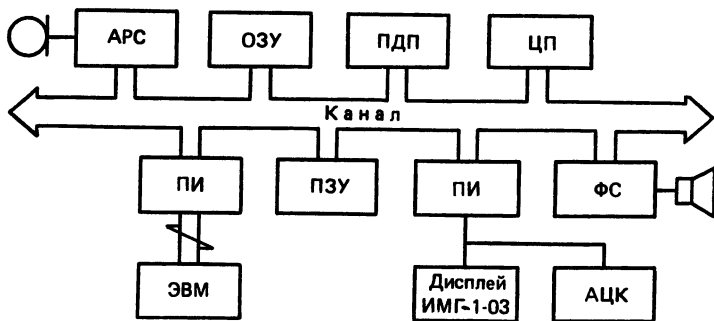


Рис. 6.2. Структурная схема устройства МАРС-1

Система МАРС-1 может работать в основных, вспомогательных и сервисных режимах. Основными режимами являются распознавание и синтез (режим «Работа»), вспомогательными — подготовка словаря команд, обучение, коррекция. К сервисным режимам относятся режим контроля, тестирование, запись и считывание сформированного словаря на внешний носитель. Выбор основного режима осуществляется системой программой в процессоре работы устройства или пользователем с помощью голоса или клавиатуры.

Работа МАРС-1 в режиме распознавания возможна, только если в ОЗУ имеется словарь команд и их эталонов. Словарь может быть сформирован пользователем в режимах подготовки словаря и обучения или переписан в ОЗУ с внешнего носителя информации. Число команд, которые могут быть включены в словарь, зависит только от емкости ОЗУ, выделенной для хранения эталонов команд. Виртуальное увеличение числа команд достигается использованием системного диспетчера памяти эталонов, определяющего из общего словаря подсловарь эталонов, с которыми осуществляется сравнение реализации на каждом конкретном шаге диалога.

В режиме контроля распознавания решение инициируется произнесением распознанной команды или словами «Вас не понял», «Повторите» и «Повторите громче» в зависимости от того, какая из гипотез была принята в качестве решения. Результат распознавания выводится также на экран дисплея. В режиме «Работа», если команда распознана правильно, то ее семантический код (название команды) и порядковый номер в словаре передаются на исполнение при произнесении следующей команды или командой «Исполнить». Неверный результат распознавания аннулируется командой «Ошибка».

В программу синтеза речи в качестве подпрограмм введены фоновое транскрибирование текста и преобразование число-числительное. В режиме контроля синтеза при вводе с клавиатуры синтезируемого текста организован пословный контроль правильности ввода информации: после ввода каждого слова по знаку пробела введенное слово произносится синтезатором. После ввода знака конца предложения (точка и др.) произносится вся фраза (метод синтеза описан в § 5.4).

Режим подготовки словаря предназначен для формирования словаря названий команд, которые пользователь считает необходимым в него включить. Названия команд вводятся с клавиатуры. В режиме обучения осуществляется формирование словаря эталонов тех команд, названия которых внесены в словарь названий. Имеется возможность визуального просмотра словаря команд, что позволяет убедиться, включены ли в словарь необходимые пользователю команды.

Большинству современных устройств речевого ввода информации свойственны два недостатка, серьезно препятствующие их практическому использованию: высокая чувствительность канала речевого ввода к мешающему воздей-

ствию динамических помех и необходимость произнесения речевых команд с паузами не меньше заданной длины. Модификация речевого терминала МАРС-М в значительной мере свободна от указанных недостатков благодаря использованию в режиме распознавания принципа обнаружения слов заданного словаря по мере их реализации во входной фразе, а в режиме обучения — принципа формирования эталонов слов, отображающих закономерные вариации их слитного произнесения [Дегтярев и др., 1987].

6.3. Перспективные зарубежные системы речевого общения

Из всего разнообразия зарубежных промышленных СРО рассмотрим системы, которые, на наш взгляд, являются наиболее перспективными в плане реализации в них как новых теоретических моделей, так и достигнутых показателей качества синтеза и распознавания речи

Примером высококачественного синтезатора, с помощью которого можно воспроизвести близкую к естественной речь различного темпа и оттенков непосредственно по тексту неограниченного словаря, является устройство DECtalk [Dagrow, 1984]. Устройство DECtalk (см. табл. 6.1) представляет собой блок размером 10×45×30 см, на задней панели которого расположены разъемы для подключения ЭВМ, видеотерминала и печатающего устройства, различных устройств вывода речевой информации, в том числе телефона, индикатора на светодиодах, регулятора громкости. Высокое качество синтезатора DECtalk обусловлено большой библиотекой используемых правил, возможностью выбора словаря по желанию пользователя, качеством аппаратурных средств для преобразования речи. Пользователь может выбрать семь различных голосов, в том числе стандартные женский, мужской и детский, а также низкий мужской и голос пожилого человека.

В устройстве DECtalk процесс преобразования текстовой информации в речевую разделяется на три уровня. На первом слова преобразуются в цифровую форму в соответствии с кодом ASCII и разбиваются на фонемы. При этом используются набор правил преобразования букв в звуки и два словаря. Один содержит 6000 слов, другой — около 150 специфических терминов, иностранных слов и аббревиатур, вводимых самим пользователем. Преобразования первого уровня начинаются с поиска эталонных слов, соответствующих вводимому орфографическому тексту, в большом словаре. Если эталоны найдены, то текст, преобразованный в фонемную форму, сразу передается на второй уровень обработки. Если эталоны не найдены ни в одном из словарей, то вводимый текст обрабатывается в соответствии с правилами преобразования букв в звуки и затем передается на второй уровень. На втором уровне производятся считывание фонем, синтаксический анализ, определяются интонация, продолжительность и ударение в словах, выполняются акустические расчеты. На третьем уровне выполняются окончательные преобразования входной текстовой информации и синтез речи. Синтезированные цифровые сигналы передаются со скоростью 120 Кбит/с в стандартный цифроаналоговый преобразователь (ЦАП). В синтезаторе DECtalk используются эвристические правила для оценки влияния окружающих слов на произношение, интонацию, продолжительность и ударение в данном слове.

В состав синтезатора DECtalk входят универсальный 16-разрядный микропроцессор Motorola-68000, применяемый на первых двух уровнях, и специальный процессор для обработки цифровых сигналов TMS-32010 фирмы Texas Instruments, используемый на третьем уровне. Микропроцессор Motorola-68000 осуществляет обращение к ЗУПВ емкостью 64 Кбайт и ПЗУ емкостью 256 Кбайт. Более половины ПЗУ отведено под большой словарь. Наряду с ЦАП в состав синтезатора DECtalk входят также устройство ввода обратного магазинного типа для TMS-32010, фильтр для подавления паразитных сигналов на выходе ЦАП, интерфейс для подключения телефонной линии, усилитель аналогового сигнала, интегральные схемы для подключения интерфейса RS-232.

Программное обеспечение для синтезатора DECtalk практически аналогично программному обеспечению для существующих алфавитно-цифровых терминалов, работающих в стандарте ASCII, за исключением модулей, определяющих формат данных. Поэтому создание программного обеспечения не вызывает трудностей, что обеспечивает синтезатору DECtalk широкие области применения.

Создание перспективных моделей распознавания речи связано со стремлением достичь высокой надежности при работе с большими словарями (свыше 1000 слов) в режиме слитного произнесения. Наиболее близко к такому идеалу подходит системы KVS-3000 [Kurzweil, 1985], словарь которой содержит 1000 слов, а точность распознавания (см. табл. 6.2) 95 %. Для подготовки устройства к работе весь набор эталонных слов должен быть произнесен трижды, поэтому память устройства KVS-3000 рассчитана на хранение 3000 слов. Среднее время распознавания около 500 мс. В состав устройства KVS-3000 входят три платы процессоров: воспринимающего звуковой сигнал и преобразующего его в цифровую форму; выполняющего операции распознавания; координирующего работу первых двух. Устройство выполнено на базе микропроцессора Motorola-68000, ОЗУ емкостью 640 Кбайт и разработанных фирмой Kurzweil AI цифровых фильтров.

Программное обеспечение KVS-3000 способно адаптироваться к изменяющемуся в процессе работы словарю эталонов. Подключая дополнительные блоки памяти, можно расширить возможности KVS-3000 до распознавания 5000 слов в системах, ориентированных на голос одного диктора. Устройство KVS-3000 может быть использовано для ввода данных в системах преобразования речи в текстовую информацию, рассчитанных на работу со словарем 10 000 слов.

Фирма Verbex (США) разработала мощную систему распознавания непрерывной речи — модель Verbex-3000 (см. табл. 6.2), позволяющую распознавать предложения любой длины, состоящие из отдельных или слитно произносимых слов [Ross et al., 1984]. Эта система ориентирована на голос одного диктора, при смене диктора необходима замена ОЗУ для хранения эталонов слов. Модель Verbex-3000, построенная по архитектуре «звезды», включает 2—4 таких речевых процессора и может оперировать 120—360 словами. Каждый процессор содержит 3У емкостью 0,25 Мбайт для хранения данных, а также 3У для хранения 4000 микрокодовых инструкций, каждая длиной 64 бит. Кроме того, гибкость конструкции системы, обеспеченная развитой структурой элементов сопряжения, позволяет использовать память центральной ЭВМ. Быстродействие речевого процессора 5 млн. опер./с.

Микропроцессорная архитектура системы позволяет производить операции с высоким быстродействием, что является необходимым условием для выполнения алгоритма распознавания непрерывной речи. Процесс распознавания, который заключается в подборе последовательности эталонных слов, максимально соответствующей произнесенной пользователем, осуществляется ДП-методом. Входные речевые сигналы усиливаются, фильтруются и кодируются на частоте 16 кГц аудиопроцессором, а затем передаются в управляющий процессор с интервалом 10 мкс. Управляющий процессор распределяет сигнал между речевыми процессорами, которые выделяют его фонетические характеристики. Разработчики системы определили, что для обеспечения точности распознавания 99 % достаточно 16 таких характеристик. Процесс распознавания в управляющем процессоре и поиск эталона производятся одновременно, т. е. когда процессор «улавливает» окончание речи.

Во избежание влияния на окончательный результат распознавания ошибок, возможных на первом этапе, система постоянно проверяет множество параллельных гипотез, сравнивая ввод данных с грамматически обозначенным набором всех нормальных произношений. Ответ задерживается до момента проверки всех гипотез и совпадения произнесенного с эталоном.

6.4. Тенденции применения средств речевого общения

До последнего времени большинство работ, связанных с созданием СРО, базировалось на идее формирования акустических эталонов речевых элементов разных уровней (фонем, слогов, слов). При этом распознавание было реализовано как обнаружение и идентификация в потоке речи этих акустических эталонов. В настоящее время стало очевидным, что такой подход малоперспективен из-за значительной вариативности акустических характеристик речевого сигнала, связанной как с лингвистическими (эффекты коартикуляции, редукции, ассимиляции), так и с экстралингвистическими факторами (индивидуальные особенности голосов дикторов, различия в их произносительных навыках, состояние проводящей среды и др.) [Klatt 1983]. Особенно большую вариативность имеют акустические характеристики фонем. Поэтому в большинстве промышленных систем распознавания речи отказались от фонемного анализа и приняли слово в качестве минимального распознаваемого элемента. Такой подход, хотя и дал на определенном этапе речевых исследований зримые результаты, оказался неплототворным при решении сложных задач распознавания, таких, как распознавание речи без подстройки под диктора или создание систем с большими словарями. Это заставило вернуться к идее фонемного анализа речи на новом качественном уровне, который предполагает моделирование различных аспектов процесса переработки информации человеком при восприятии речи, использование знаний, накопленных в лингвистике, теории восприятия и психологии. При таком подходе одной из кардинальных задач распознавания речи является задача обнаружения и моделирования тех механизмов восприятия речи человеком, которые обеспечивают устойчивость и стабильность воспринимаемых речевых элементов в условиях их огромной акустической вариативности.

Все большее распространение получает подход к решению задачи распознавания речи, связанный с чтением динамических спектрограмм неизвестного речевого сигнала фонетистом-экспертом. Интерес к экспериментам по чтению спектрограмм связан, во-первых, с тем, что они демонстрируют богатство фонетической информации, заключенной в речевом сигнале, представленном в виде трехмерной (частота — время — интенсивность) спектральной картины, а во-вторых, с тем, что они позволяют воплотить знания и процедурные навыки эксперта в системах распознавания речи. Один из ведущих исследователей в этой области В. В. Зу [Zue et al., 1986] отметил, что чтение спектрограмм дает нам «доказательство существования» того, что высоконадежное фонетическое декодирование речевых спектров является достижимым, а знания, используемые в чтении спектрограмм, являются потенциально передаваемыми как людям, так и машинам. Исследования по чтению спектрограмм ведутся в разных странах, и в настоящее время главным в этих исследованиях является проблема выявления и формализации знаний и процедурных навыков экспертов. Оказалось, что это весьма сложный процесс, так как эксперты не всегда способны выразить принципы и правила, которыми они руководствуются в ходе акустико-фонетического декодирования речевых спектрограмм. Одной из существенных проблем [Stern et al., 1986] является воплощение в алгоритме правил и знаний, которые кажутся эксперту очевидными (например, сведения о том, что F_2 не может быть выше 2500 Гц) либо сложно формулируемыми (например, правило определения компактности взрыва смычных). Для преодоления этих трудностей, связанных с извлечением и записью знаний эксперта, используются экспертные системы (см. гл. 13, 14).

Экспертные системы (ЭС) отличаются от обычных компьютерных программ тем, что они могут решать задачи, не имеющие жестких алгоритмических решений, и осуществлять выводы, основанные на неполной или недостоверной информации [Rauch, 1984]. Поэтому они гораздо более приспособлены для моделирования гибкой деятельности человека, что и было использовано в речевых исследованиях. Необходимо подчеркнуть, что в этих исследованиях ЭС применяются именно как инструмент анализа для накопления базы знаний в

процессе взаимодействия системы и эксперта. Использование ЭС в качестве системы распознавания [Stern et al., 1986] неэффективно. Аналогичен подход и авторов работы [Caelen et al., 1986], в которой описана разработанная ими ЭС для выявления знаний эксперта и продемонстрировано, как полученная таким образом база знаний использована во фреймовой модели акустико-фонетического декодирования.

Вторая существенная трудность при моделировании деятельности фонетиста-эксперта заключается в том, что визуальный анализ спектрограмм, лежащий в основе декодирующих действий эксперта, не менее сложен, чем слуховая обработка речевого сигнала. При формализации чтения спектрограмм возникает сложная проблема извлечения акустических признаков, легко выделяемых зрительной системой человека (например, определение формантных треков). При разработке ЭС многие исследователи эту трудность просто обходят. Так, в работе [Stern et al., 1986] описана система, на вход которой подаются словесные описания спектрограмм, которые делает человек, а в системе, рассмотренной в работе [Zue et al., 1986], осуществляется полуавтоматическое выделение существенных для фонетического декодирования акустических признаков, дополняемых во многих случаях человеком. Однако уже из самого рассмотрения возникшей ситуации логически вытекает представление о наличии некоторого промежуточного уровня кодирования речевой информации при переходе от параметрического описания спектра к его фонетической интерпретации. Более того, необходимость разработки принципов промежуточного описания речевых сигналов выделяется в качестве центральной проблемы автоматического распознавания речи [Green et al., 1986]. Авторы этой работы отмечают, что только промежуточное описание позволяет перекинуть мост между непосредственно наблюдаемым непрерывным акустическим сигналом и дискретным лингвистическим описанием. Промежуточное представление описывает «поведение» акустических характеристик, которое в значительной степени является инвариантным при переходе от диктора к диктору и зависит главным образом от контекстного взаимодействия фонетических единиц. При этом промежуточное представление, как правило, выражается в качественном виде и описывается в терминах наличия-отсутствия определенных акустических объектов, высокого-низкого положения в спектре или сильного-слабого проявления той или иной энергетической составляющей.

Таким образом, в процессе акустико-фонетического декодирования осуществляется вначале переход от количественных измерений к качественному описанию, а затем от качественных описаний к признакам фонем [Zue et al., 1986]. Аналогичный подход предложен и в работе [Caelen et al., 1986], авторы которой считают необходимым преобразовывать базу данных, полученных в результате первичной обработки, в базу «явлений» или «сущностей», которая служит основой принятия фонетических решений. Как правило, единицы промежуточного уровня представления речевой информации называются акустическими ключами, акустическими признаками или дискрипторами.

В СССР работы по чтению спектрограмм фонетистами-экспертами, разработке правил их фонемной и словесной интерпретации проводятся с 1980 г. [Зиновьева, 1984; Зиновьева и др. 1986]. Полученные результаты позволили приступить к созданию аппаратно-программной модели анализа речи, в основу которой положены алгоритмы чтения спектрограмм [Зиновьева, 1984].

Особенности построения и использования систем переработки визуальной информации

В. В. Александров, В. А. Анисимов, Н. Д. Горский, С. Н. Мысько

7.1. Назначение, классификация и области применения

В настоящее время в различных областях науки и техники ощущается рост потребностей в переработке, анализе и отображении визуальной информации. В этом направлении принято выделять три основных типа задач [Павлидис, 1986]: собственно *обработку изображений*, когда и исходные данные, и результаты обработки представляются в изобразительной форме; *анализ* (интерпретацию, распознавание или «понимание») изображений, когда входные данные являются изображением, а результат представляется в неизобразительной форме, например в виде текстового описания наблюдаемой сцены; *синтез изображений* (машинную графику), когда на входе имеется описание (алгоритм построения) изображения, а на выходе по нему строится само изображение. Взаимосвязь трех перечисленных типов задач показана на рис. 7.1.

Обработка изображений, как уже отмечено, связана с преобразованием изобразительной информации вновь в изобразительную форму. Примером здесь может служить устранение искажений или дефектов на изображении, улучшение качества получаемой визуальной информации путем повышения контраста, подчеркивание контуров объектов и т. д. Основные направления обработки изображений выделены в [Горский, 1989].

Задачей анализа изображений является получение из изображения, поданного на вход системы, неизобразительного описания. Это описание может быть различных уровней общности — от простого указания номера или имени класса, к которому относится анализируемое изображение, до подробной характеристики наблюдаемой сцены с указанием отдельных объектов и отношений между ними. В последнем случае обычно говорят о «понимании» изображений. Некоторые авторы называют анализ изображений распознаванием, хотя, на наш взгляд, этот термин имеет более узкое значение, относясь преимущественно к идентификации отдельных объектов на изображениях. К типичным задачам анализа можно отнести распознавание рукописных или печатных знаков, дешифрацию аэрофотоснимков, анализ наблюдаемых сцен.

Синтез изображений обычно отождествляют с машинной графикой, хотя последний термин в буквальном понимании стал несколько архаичен. В настоящее время синтезируются на ЭВМ отнюдь не только «графические» картины, наоборот, синтез все больше претендует на создание полноцветных реалистических изображений по их описаниям в неизобразительной форме. Сюда относятся и системы имитации визуальной обстановки на тренажерах, и системы геометрического моделирования в САПР, и системы компьютерного киноискусства. Наряду с этим сохраняются и традиционные приложения машинной гра-



Рис. 7.1. Взаимосвязь направлений цифровой обработки изображений

фики: вывод информации на экран дисплея в виде диаграмм, графиков; построение чертежей и т. д. *Интерактивной графикой* называют направление, предполагающее не только синтез изображения машиной, но и ввод пользователем в систему данных в графической форме, например, с помощью светового пера или манипулятора типа «мышь».

Системы обработки и анализа видеоданных, которые обычно работают со входной информацией в виде полутоновых или многоспектральных изображений, в нашей стране обычно объединяют под названием *автоматизированных систем обработки изображений* (АСОИз) [Автоматизированные, 1986; Овсеевич и др., 1985]. В настоящее время системы, решающие достаточно сложные и «интеллектуальные» задачи анализа изображений, принято выделять в отдельный класс систем анализа изображений (САИ). Системы, задачами которых являются синтез изображений и обработка графической информации, называют *системами машинной графики* (СМГ); употребляется также термин «системы обработки графической информации» (СОГИ) [Автоматизация, 1986; Методы, 1985; Павлидис, 1986]. Следует отметить, что такое деление достаточно условно, и за последнее время наметилась устойчивая тенденция к сближению этих систем. Все чаще АСОИз оперируют графической информацией, решая в том числе и задачи синтеза, а СМГ применяются для построения полутоновых изображений с привлечением различных методов их обработки.

Одно из направлений наиболее широкого практического применения АСОИз — обработка результатов дистанционных исследований. Среди многочисленных развивающихся прикладных областей здесь можно упомянуть: геологию (исследование и поиск природных и ископаемых ресурсов); сельское и лесное хозяйство (обнаружение аномалий, слежение за состоянием участков, предсказание урожая); анализ состояния окружающей среды и климата; метеорологию (предсказание погоды, анализ воздушных течений и температурных полей); гидрологию и океанографию (наблюдение за водными ресурсами, течениями, распределениями температуры, планктона, льдов); картографию (составление фотোগрафических и других карт, накопление картографических данных, территориальный анализ); слежение за объектами на земной поверхности (транспорт, судовождение и т. д.).

Быстро развивается направление обработки биомедицинских данных. Здесь АСОИз находят применение в микробиологии и цитологии (анализ снимков с микроскопов, анализы крови, хромосомный анализ), рентгенологии, ультразвукологии, инфраскопии (обработка двух- и трехмерных изображений внутренних органов, анализ распределения температур тела, томография), сравнительных исследованиях (идентификация отпечатков пальцев, идентификация состояний больных). Недостаточно развито пока применение АСОИз в производственных системах, однако потребности автоматизации управления и производства должны привести к быстрому росту числа применений АСОИз в таких областях, как материаловедение (обнаружение дефектов, анализ и предсказание свойств материалов, структурный анализ), сборочные работы (системы технического зрения для роботов и гибких автоматизированных производств), системы ориентации и анализа обстановки в реальном времени (для различных видов транспорта, летательных и космических аппаратов).

Основными областями применения систем машинной графики являются: автоматизация проектирования (построение чертежей, проекций, планов, двух- и трехмерных визуальных моделей деталей и объектов); дизайнерские работы в таких областях, как автомобиле- и самолетостроение, архитектура, пошив одежды; картография (составление, корректировка, хранение и генерация карт); автоматизация исследований (отображение в визуальной форме результатов экспериментов, вычерчивание графиков, гистограмм, диаграмм и т. д.); организация эффективного взаимодействия человека с ЭВМ (графические языки, интерактивная графика); синтез картин и сцен (для тренажеров самолето- и автовождения, в машинной живописи, в компьютерных играх); синтез динамических картин (компьютерная мультипликация и кинематография).

Степень «интеллектуализации» АСОИз и систем машинной графики может

быть различной в зависимости от уровня представления и использования в этих системах знаний специалистов по обработке изображений и специалистов предметных областей. Она колеблется от возможности применения набора утилит (операций) обработки, анализа или синтеза изображений, которые в грубом приближении можно интерпретировать как записанные в процедурной форме записи специалистов, до поддержки специальных баз знаний в развитых экспертных системах анализа изображений.

В этой связи в настоящем разделе при последующем изложении рассматриваются системы трех классов:

традиционные «малоинтеллектуальные» АСОИЗ, без понимания состава, функций и особенностей построения которых трудно представить себе уровень развития интеллектуальных систем;

системы анализа видеоданных, основанные на знаниях, которые образуют специальный класс экспертных систем анализа изображений и которым уделено ниже преимущественное внимание;

интеллектуальные системы машинной графики.

7.2. Автоматизированные системы обработки изображений

Исследовательские АСОИЗ создаются обычно как «универсальные», не привязанные жестко ни к какому классу видеоданных. В то же время данные конкретных предметных областей имеют свою специфику и особенности, что необходимо требует проблемной ориентации АСОИЗ, разрабатываемых для прикладных применений. Первые такие системы были специализированными, предназначенными для решения конкретных узких задач с помощью ограниченного набора алгоритмов. С развитием методов обработки изображений, удешевлением аппаратуры, появлением новых технологий (в первую очередь СБИС) и технических средств (микропроцессоров, интеллектуальных видеотерминалов, памяти большой емкости и т. п.) проявилась устойчивая тенденция к переходу от систем специализированных к системам проблемно-ориентированным, призванным решать некоторый класс задач в рамках данной предметной области.

Проблемно-ориентированные прикладные АСОИЗ используют, как правило, достаточно богатый арсенал различных методов и средств обработки данных, анализируют различную входную информацию, активно используют возможности человека. Поэтому в таких системах должны быть различные функциональные блоки и подсистемы, обеспечивающие не только решение собственно задач обработки изображений, но и ввод и обработку других типов данных и знаний, накопление, хранение и отображение результатов, организацию взаимодействия с пользователем и другими системами. Наличие разнородных компонентов, работающих совместно для достижения общей цели, есть признаки превращения современных АСОИЗ в *интегрированные системы* [Александров и др., 1984б]. Интеграция является одной из главных тенденций развития современных АСОИЗ, поэтому рассмотрим ее направления и особенности более подробно.

Прежде всего АСОИЗ уже нельзя рассматривать обособленно от ее окружения — средств сбора и передачи видеоданных, систем обработки и анализа данных других типов. АСОИЗ, рассматриваемая как элемент некой метасистемы, перестает быть элементом изолированным, приобретает богатые информационные связи с другими элементами. Это интеграция внешняя. Показательным примером служит здесь японский проект создания ЭВМ пятого поколения. Согласно этому проекту система распознавания образов и изображений (наряду с системами машинного перевода, ответов на запросы, понимания специализированного языка и решения прикладных задач) является базовой в обеспечении интеллектуального интерфейса человек-ЭВМ, при этом все упомянутые системы должны работать параллельно и одновременно над единым потоком данных, дополняя друг друга. Интеграция приводит к появлению в составе АСОИЗ развитых средств коммуникации, подсистем накопления и хранения как исходных данных, так и промежуточных и окончательных результатов (рис 7.2).

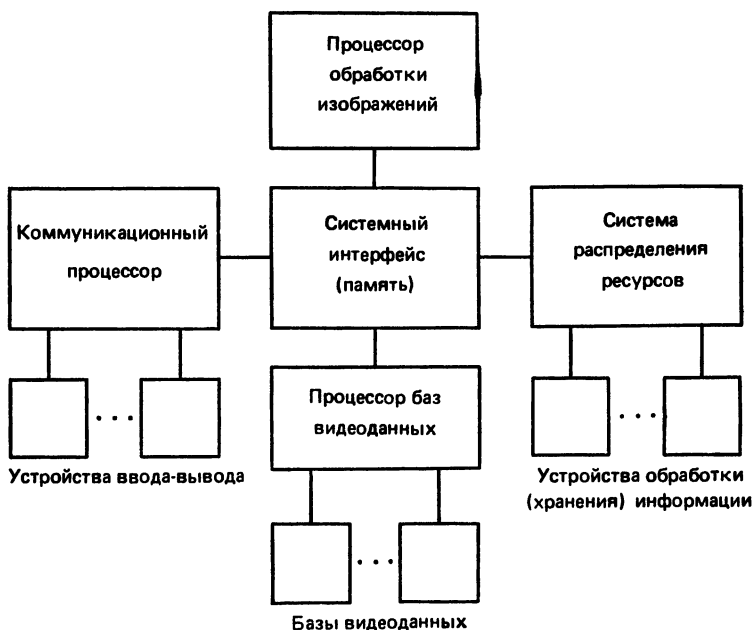


Рис. 7.2. Структура интегрированной АСОИз

Внутренняя интеграция затрагивает различные компоненты АСОИз: средства и методы обработки, сами обрабатываемые данные.

Интеграция средств обработки означает объединение в системе различных универсальных и/или специализированных устройств, имеющих возможность параллельной работы над данными. Например, в системе PUMPS [Briggs, 1982] интегрированы процессоры обработки изображений, коммуникации и машина базы видеоданных. В других системах [Александров и др., 1985] обработка и анализ изображений ведутся параллельно с помощью независимых специализированных процессоров, каждый из которых выполняет определенные операции: фильтрацию, вычисление площади объектов, вычисление периметра, сегментацию и т. д. Часто имеется отдельный процессор для численных расчетов, процессор внутренних обменов и процессор интерфейса. Практически все новые АСОИз снабжены развитыми средствами визуализации данных, включая полутоновые, графические и стереоскопические (трехмерные) устройства отображения.

Интеграция методов обработки означает наличие в системе функциональных подсистем и обрабатывающих алгоритмов для решения широкого круга задач и достижения результатов различными путями (возможно, параллельными). При этом к результату и приходят различными доступными путями, применяя разные методы к одним и тем же данным и интегрируя (на основе опыта и интуиции исследователя или заложенных в систему знаний) получаемую выходную информацию.

Работа по анализу изображения становится своего рода многократным экспериментированием с исходными данными. Характерной чертой такого эксперимента является совместное применение методов обработки изображений, математической статистики, искусственного интеллекта, анализа данных и других, причем в обработку по возможности вводятся все доступные данные,

включая промежуточные результаты. Появляется необходимость организации эффективного обмена данными в системе, создания развитых средств хранения и обеспечения доступа к данным (включая базы видеоданных). В настоящее время это уже характерно для многих АСОИЗ, применяемых в дистанционных исследованиях [Tamura et al., 1984; Александров и др., 1987].

Ключевым звеном АСОИЗ обычно считается процессор(ы), производящий обработку изображений. В первых системах в качестве обрабатывающего процессора часто применялись крупные универсальные ЭВМ, работающие в режиме разделения времени. Это решение оказалось малоэффективным, поскольку и системы команд, и операционные системы таких ЭВМ не ориентированы на задачи обработки изображений. В архитектуре АСОИЗ 70-х годов наряду с появлением специализированных процессоров обработки изображений получило широкое распространение монопольное использование средних и малых ЭВМ, ориентация которых на решение задач видеоданных достигалась разработкой соответствующего программного обеспечения. В настоящее время широкое распространение получили так называемые рабочие станции обработки изображений и системы, построенные на базе персональных ЭВМ [Weston et al., 1987; Ramo et al., 1986].

Однако наличие одного процессора редко удовлетворяет требованиям по производительности системы, особенно при больших потоках обрабатываемых данных. Крупные АСОИЗ строятся, как правило, на основе специальных многопроцессорных конфигураций для параллельной обработки элементов одного или нескольких изображений, при этом упомянутые средние или малые ЭВМ обычно переходят в разряд управляющих [Горский, 1986; Fountain, 1986].

Другими архитектурными компонентами АСОИЗ являются прежде всего устройства ввода видеоданных в систему (ТВ-камеры, дальномеры, малокадровые сканеры, фототелеграфные аппараты и т. д.), устройства отображения (ТВ-мониторы, растровые и векторные дисплеи, устройства вывода на твердые носители — бумагу и фотопленку, графопостроители), запоминающие устройства (на гибких и жестких магнитных дисках, оптических дисках; устройства расширения оперативной памяти для хранения изображений). Достаточно типичная конфигурация недорогой АСОИЗ, работающей под управлением персональной ЭВМ [Ramo et al., 1986], показана на рис. 7.3. Технические характеристики упомянутых компонентов АСОИЗ и другие примеры можно найти в [Александров и др., 1985, 1987; Гиммельфарб, 1985; Автоматизированные, 1986; Обработка, 1984].

В настоящее время в интегрированных системах аппаратно реализуются не только параллельные процессоры обработки изображений, но и другие подсистемы, ранее традиционно реализуемые программным путем. Прежде всего здесь следует упомянуть средства доступа к различным видам памяти и средства, поддерживающие принятие решений. Особенно большую пользу приносят такие реализации при решении задач распознавания образов, интерпретации видеоданных, анализа сцен. Фактически аппаратные средства здесь поддерживают функционирование базы данных и базы знаний, но на качественно ином уровне. Основные решаемые задачи для базы видеоданных — это поиск эталонных объектов или их отдельных частей (видеоприсказов), поиск необходимых изображений по их идентификаторам. Чтобы решить эти задачи, разрабатываются специализированные процессоры баз видеоданных и массивы ассоциативной памяти с возможностью контекстного доступа к информации (адресация по содержанию).

Сейчас ЭВМ фирмы DEC и совместимые с ними отечественные ЭВМ серий СМ и «Электроника» стали в некоторой степени стандартными для систем обработки видеоинформации. По этой причине общее программное обеспечение этих ЭВМ часто находят применение в АСОИЗ. По оценкам [Tanimoto, 1982], наиболее распространенными операционными системами (ОС) для АСОИЗ 80-х годов являются VMS (VAX) и UNIX RSX-11. Под управлением этих ОС в АСОИЗ используются различные языки высокого уровня, среди них Фортран, Паскаль, Си. С широким распространением персональных ЭВМ и «интеллек-

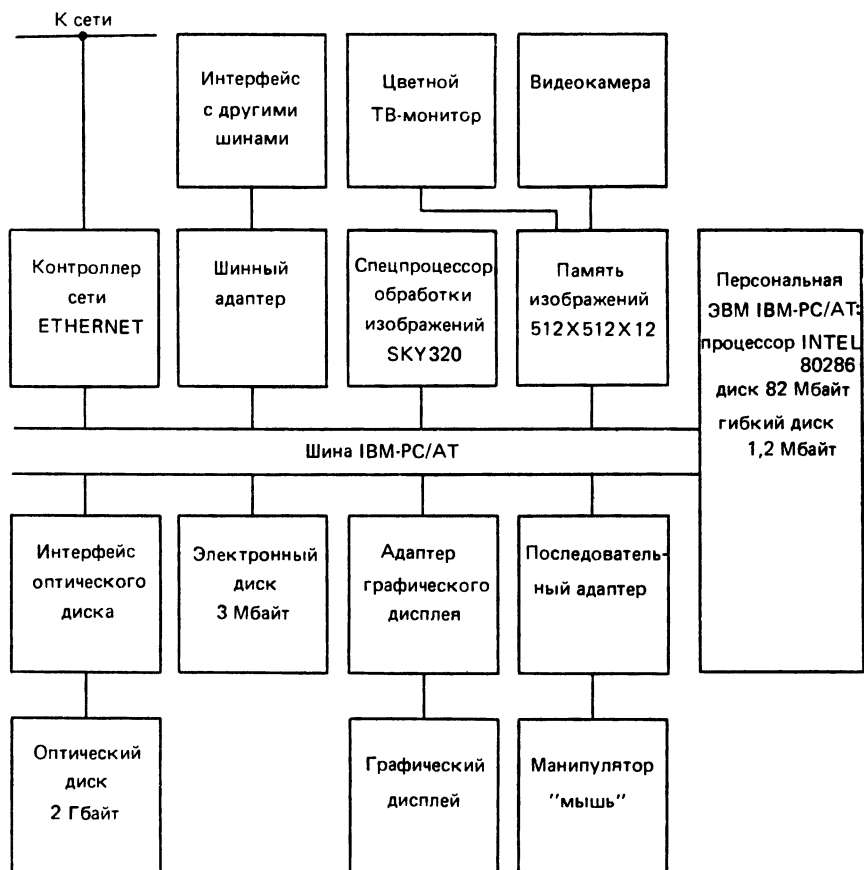


Рис. 7.3. Архитектура типичной АСОИЗ, основанной на персональной ЭВМ

туализацией» АСОИЗ все большее значение приобретают ЛИСИП и Пролог как средства поддержки и разработки подсистем без знаний и интеллектуального интерфейса. Для этих ЭВМ характерно применение таких операционных систем, как MS/DOS (высоких версий), UNIX.

Использование языков и техники параллельного программирования в АСОИЗ, по-видимому, будет расширяться, однако это расширение коснется скорее синхронных многопроцессорных систем, где распараллеливание вычислительного процесса исходит от программиста, а не потоковых систем.

Стандартные наборы операций универсальных языков программирования высокого уровня не могут обеспечить специфических потребностей пользователей АСОИЗ, поэтому одно из основных направлений в создании программного обеспечения обработки видеоданных — это разработка специализированных программных систем. Разработчики здесь идут двумя путями. Во-первых, расширяя существующие универсальные языки, создают специализированные языки программирования. Другим путем часто идут разработчики рабочих станций обработки изображений, создавая библиотеки программ или целостные программные системы, предназначенные для решения определенных задач конеч-

ных пользователей и написанные на одном из стандартных языков. Этим обеспечивается переносимость программных продуктов, что во многих случаях становится решающим фактором для пользователей.

Важное направление развития программного обеспечения АСОИЗ — создание эффективных систем хранения и поиска видеоданных (баз и банков видеоданных). Потребность в базах видеоданных ощущается прежде всего в области дистанционных исследований, поэтому именно здесь можно проследить тенденции развития программного обеспечения этих систем. От языков со встроенными функциями доступа довольно быстро произошел переход к интерактивным средствам манипулирования базой, а программы обработки запросов влились в системы поддержки и управления базой. Почти все существующие базы видеоданных — смешанные в смысле хранения как видео-, так и текстовой информации; характерной их особенностью является привязка содержания текста к элементам (объектам) изображения.

Все чаще в состав программного обеспечения АСОИЗ включаются и средства ведения баз знаний. Пока это преимущественно специализированные подсистемы для использования базы знаний о конкретной предметной области (например, о поиске месторождений определенного типа на аэрофотоснимке), но в скором времени можно ожидать появления инструментальных средств для разработки и наполнения знаниями экспертных систем анализа визуальной информации. С превращением АСОИЗ в системы для решения реальных задач повышается внимание к проблеме эффективного взаимодействия системы с конечным пользователем. Результат этого — быстрое развитие интерактивных средств общения с системой, создание «дружественных» пользователю прикладных программ, разработка языков взаимодействия командного типа и языков, основанных на выборе одной из предлагаемых системой альтернатив (меню, таблицы решений, spread sheet — крупномасштабная электронная таблица), а также развитие средств обучения, описания состояния и возможностей системы.

7.3. Системы анализа изображений

Спектр решаемых задач в существующих САИ достаточно широк: от локализации простейших объектов на фотографиях до содержательной интерпретации изображений. Тем не менее можно выделить три основных класса задач, решаемых САИ, которые определяют их различное назначение [Александров и др., 1986].

1. Распознавание объектов. В системах этого типа решаются задачи локализации определенных объектов в поле изображения, измерения заданных параметров этих объектов, принятия решения по полученным результатам измерений. Примерами могут служить системы медицинской диагностики, где необходимо локализовать участки изображения, указывающие на отклонение от нормы, определить степень отклонения; системы контроля за протеканием процессов (например, сгорания топлива в камере котельной); системы локализации дефектов печатных плат и др.

2. Содержательная интерпретация изображений. К системам, в которых решается эта задача, относится большинство САИ. Сюда входят системы управления движением робота по поступающей видеoinформации, интерпретации аэрофотоснимков, анализа снимков и сопоставления их с картой местности и др. В отличие от систем первого типа в данном случае задача заключается в полной содержательной интерпретации снимка, «понимании» отношений, связывающих объекты реального мира, представленные своими проекциями на двухмерном изображении.

3. Получение справочной информации в архивах видеоданных. Задачей этих систем является выдача информации пользователю о наличии объектов определенного типа на изображениях, хранящихся в архиве видеоданных, его параметрах, отношений к другим объектам и т. п. Запрос может быть представлен как текстовое описание объекта или как видеобраз. К системам та-

кого типа относятся различные информационно-поисковые системы в картографии, метеорологии, криминалистике и т. п.

Для решения указанных задач САИ должны обеспечивать выполнение ряда функций, среди которых основными являются следующие.

1. Хранение и поиск формализованных знаний экспертов предметной области и обработки изображений. Формализованными знаниями являются установленные факты, закономерности, количественные соотношения, видеоданные (эталоны, прототипы объектов), а также правила, определяющие последовательность и состав операций над данными для достижения заданных целей (например, для получения проекции трехмерного тела на плоскость).

2. Выполнение операций обработки изображений (контрастирование, устранение шума, выделение контурных линий и т. п.).

3. Выполнение операций логических выводов на основе формализованных знаний и результатов обработки видеоданных, построение и проверка гипотез.

4. Анализ промежуточных результатов работы системы и принятие решений о дальнейших действиях, невозможности решения задачи, необходимости дополнительной информации и др.

5. Анализ запросов пользователя, поддержка диалога, документирование действий системы и формирование пояснений пользователю.

6. Пополнение баз знаний и видеоданных системы новыми сведениями, данными.

Конкретная САИ не обязательно выполняет все перечисленные функции. Их набор определяется в первую очередь назначением системы и может быть ограниченным для относительно простой системы локализации «подозрительных» мест на рентгеновском снимке или полным для более сложной системы интерпретации изображений сцены, сопоставления его с картой и принятия решения о выборе маршрута движения робота.

Современные системы анализа изображений довольно разнообразны по своей структуре и организации взаимодействия ее составных частей. Ниже будет рассмотрен весьма популярный в последнее время подход к реализации САИ в виде экспертной системы (ЭС) анализа изображений (ЭСАИ). Под ЭСАИ понимают комплекс технических и программных средств, обеспечивающий решение проблемно-ориентированных задач, требующих проведения обработки и анализа видеoinформации, подобно тому как это делают эксперты конкретной предметной области. ЭСАИ предполагает активное использование формализованных знаний, проведение обработки и анализа изображений на основе одной или нескольких моделей, адекватно отражающих с точки зрения экспертов процессы и взаимоотношения между реальными объектами конкретной предметной области.

На рис. 7.4 приведена упрощенная структурная схема ЭСАИ в ее наиболее полном варианте, соответствующем системе содержательной интерпретации изображений. Из рисунка видно, что ЭСАИ объединяет в себе как средства традиционных систем обработки изображений, баз видеоданных, так и аппарат ЭС: базу знаний, блок управления и принятия решений (см. гл. 13).

Подсистема хранения и обработки изображений содержит программные и аппаратные модули для выполнения различных операций над видеоданными, поиска изображений по видео- и текстовому запросу, а также архив изображений. Эта часть ЭСАИ в целом не отличается от программных и технических средств, используемых в традиционных системах обработки изображений. Важной особенностью подсистемы является модульный принцип построения, благодаря которому система может собирать технологические цепочки последовательностей операций для решения конкретных задач. Например, задача выделения контурных линий может решаться следующей последовательностью операций: подавление шума; контрастирование; применение градиентного оператора для выделения границ; отслеживание контурных линий, величина перепада яркости которых превышает заданный порог. Решению этой задачи могут предшествовать операции поиска заданного изображения в базе видеоданных.

В базе знаний системы хранятся формализованные знания специалистов

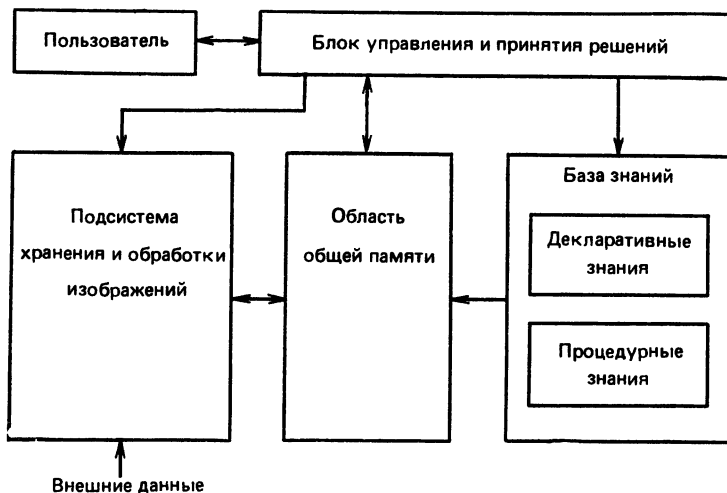


Рис. 7.4. Структура экспертной системы анализа изображений

предметной области и обработки изображений. Условно в ней можно выделить два компонента: декларативные и процедурные знания:

Декларативные знания отражают сведения об объектах, их составе, взаимном расположении, характере появления, описывают отношения между объектами и т. п. Авторы ЭСАИ [Rosenthal et al., 1984] выделяют три типа описания объектов: контекстное, визуальное и характеризующее повторяемость объекта в сцене. Так, маркировочную осевую линию следует искать на изображении шоссе, линия должна быть светлее фона, прямоугольной формы, по характеру повторяемости единичной. В другой ЭСАИ [Levine et al., 1981] декларативные знания описывают пространственные соотношения объектов (дерево ниже облаков, дверь дома ниже крыши), отношения сравнения (разброс значений яркости изображения неба не превышает определенного порога). В системе анализа снимков аэропорта [McKeown et al., 1983] хранятся сведения о том, что здания ангаров и аэропорта самые большие в окрестности, взлетные и посадочные полосы прямолинейные и т. п. Следует отметить, что для большинства ЭСАИ существенная часть декларативных знаний связана с пространственными, количественными, классификационными отношениями объектов, а такие отношения, как временные, причинно-следственные, пока еще слабо используются.

Особенно следует отметить отношение типа обобщение-спецификация, используемое в большинстве современных ЭСАИ и отражающее иерархию абстракций — последовательности представлений объектов от примитивов (перепавов яркостей, однородных участков и т. п.) до абстрактных понятий (здание, дорога, автомобиль и т. п.). Построение иерархий абстракций является, по мнению многих исследователей [Баллард и др., 1987], ключевым моментом воспроизведения процесса восприятия. Считается, что окончательная интерпретация изображения должна выполняться системой на верхних понятийных уровнях иерархии.

Второй основной частью базы знаний являются процедурные знания, показывающие, какие действия, в какой последовательности нужно предпринимать системе при выполнении определенных условий, отражающих сложившуюся ситуацию анализа данных, формализованных знаний и результатов обработки изображений. Процедурные знания используются для выполнения двух функ-

ций: пополнения описания сложившейся ситуации новыми фактами из базы знаний; построения гипотез и выбора конкретных действий по преобразованию полученных результатов и переходу к другому состоянию решения задачи. Например, по локализованному на снимке объекту можно определить смежные с ним объекты, их предполагаемые размеры, форму (полное описание и генерация гипотез), провести операции обработки с целью проверки предположений (выбор действий), после чего принять решения о состоятельности выдвинутых гипотез.

Процедурные знания обычно представляются в виде правил, разделенных на две части: условие выполнения действия и назначаемое действие (последовательность действий) в случае выполнения условия. Например, правило может назначить операцию соединения сегментов контурной линии, если разрыв между ними достаточно мал. Интерпретация объектов может считаться состоятельной (действие — установление признака состоятельности интерпретации), если выполняются все предусмотренные между ними отношения, и т. д.

Область общей памяти, называемая также доской объявлений (blackboard) или базой данных, является частью системы, обеспечивающей передачу результатов работы всех подсистем друг другу. Обычно область общей памяти представляет собой реляционную базу данных, куда в ходе решения задачи записываются промежуточные интерпретации участков изображений (карты, платы интерпретации), установленные факты, количественные соотношения и другие результаты. В область общей памяти могут заноситься также пояснения действий ЭСАИ, с тем чтобы пользователь мог при необходимости оценить рациональность поведения системы, выявить причину принятия того или иного решения.

Блок управляющих процедур и принятия решений является координирующим центром системы, анализирующим запросы пользователя, сложившуюся ситуацию анализа, определяющим контекст и стратегию решения. Информация о промежуточных результатах анализа поступает в блок из области общей памяти. По результатам анализа состояния области общей памяти (сложившейся ситуации) блок управляющих процедур и принятия решений определяет состоятельность полученной интерпретации, генерирует гипотезы (о принадлежности объектов определенным классам, о возможности нахождения других объектов и т. п.) и «поручает» другим модулям системы выполнение конкретных действий. Окончательное решение также принимается этой подсистемой.

Процесс решения задачи содержательной интерпретации изображения является многоступенчатым, итеративным. На начальной стадии обычно выполняется предварительная сегментация изображения. Часть авторов считает, что предварительная сегментация должна быть консервативной [Levine et al., 1981] без использования каких-либо знаний. Примером может служить разбиение изображения на однородные по яркости участки путем слияния областей (без их расщепления). Другие исследователи полагают [Matsuyama et al., 1985], что сегментация всегда должна быть целенаправленной, т. е. использующей знания, и проводиться в определенных участках изображения на основе соответствующих моделей объектов.

Следующим этапом является получение начальной интерпретации результатов предварительной сегментации. Это включает в себя измерения определенных параметров (например, площадей или периметров полученных областей с одинаковой яркостью) и генерацию гипотез (с помощью сопоставления результатов измерений с информацией из базы знаний).

Все результаты интерпретации записываются в область общей памяти. Изменения состояния этой области означают, что процесс интерпретации не закончен. Блок управляющих процедур и принятия решений анализирует состояние области и вырабатывает новую последовательность действий. Процесс решения задачи обычно считается завершенным, если не происходят существенные изменения в области общей памяти (на карте интерпретации).

Обычно в ЭСАИ используются две глобальные стратегии решения задач интерпретации изображений (и также их комбинации): управление целью (на-

ываемое также управлением сверху вниз или обратным выводом) и управление данными (называемое также управлением снизу вверх или прямым выводом). В первом случае последовательность действий выбирается исходя из поставленной цели. Например, при локализованном на аэрофотоснимке объекте «здание аэропорта» ставится цель обнаружить рядом подземные пути. Такая стратегия соответствует продвижению сверху вниз по иерархии абстракций: от понятия объекта к его геометрическим примитивам на изображении. Вторая стратегия соответствует движению в обратном направлении по иерархии абстракций. Последовательность действий определяется на основе полученных результатов обработки данных. Например, система пытается интерпретировать прямоугольный участок изображения аэрофотоснимка, полученный в результате сегментации, как объект класса «здание».

Большинство разработчиков ЭСАИ придерживаются мнения, что обе стратегии должны присутствовать в решении задачи системой. Переключение стратегий остается нерешенной проблемой, так же как и нет единого мнения о том, в каких случаях система должна выбрать ту или иную стратегию.

7.4. Системы машинной графики

Машинная графика стала обязательной частью любой интеллектуальной системы, реализованной на ЭВМ. Это связано прежде всего с тем, что скорость переработки человеком информации, представленной в графической форме, выше по сравнению со скоростью переработки алфавитно-цифрового набора данных. Особенно заметен этот эффект на этапе восприятия информации, так как при этом максимально активизируется ассоциативное мышление человека. Наиболее ярко преимущества графического представления данных проявляются в интерактивных системах машинной графики, подразумевающих активную обратную связь с пользователем. Такие СМГ сейчас используются в САПР, АСНИ, экспертных системах и т. д. [Гардан и др., 1987; Ван Дам, 1986; Маркин и др., 1987].

Состав типичной СМГ представлен на рис. 7.5. Схематически работу системы можно представить следующим образом. На основании базы знаний и банка данных ЭВМ генерирует модель «мира». Затем эта модель преобразуется программой синтеза изображения в удобную для восприятия человеком форму. Сформированная таким образом информация выдается на устройство отображения графической информации, вызывая на нем появление соответствующего изображения, которое может быть задокументировано с помощью устройства получения твердых копий. На этапе окончательного формирования изображения в него могут включаться готовые фрагменты из банка видеоданных. Пользователь через устройство ввода информации может формировать и дополнять базу знаний и банк данных, корректировать модель «мира» и программу синтеза изображения, а также производить непосредственные операции над изображением (вырезание и перемещение фрагментов, их склейку, масштабирование, раскраску и т. д.), воздействуя через сервисные программы непосредственно на буфер устройства отображения.

Модель «мира» является центральным звеном практически всех современных СМГ. Эта модель порождает «мир», состоящий, например, из схем, элементарных частиц или реально наблюдаемых объектов. Все объекты «мира» описываются своими наборами характеризующих их параметров. По размерности моделируемого «мира» модели принято делить на двух- и трехмерные. Трехмерные модели, в свою очередь, подразделяются на каркасные («проволочные»), поверхностные и модели сплошных тел.

В каркасных моделях хранятся (генерируются) лишь координаты вершин объекта и соединяющие их ребра. При таком подходе часто возникает неоднозначность, так как возможны различные интерпретации одной и той же модели. В моделях поверхностного типа хранятся только описания поверхностей, ограничивающих данный объект. Эти поверхности могут быть плоскими, ци-

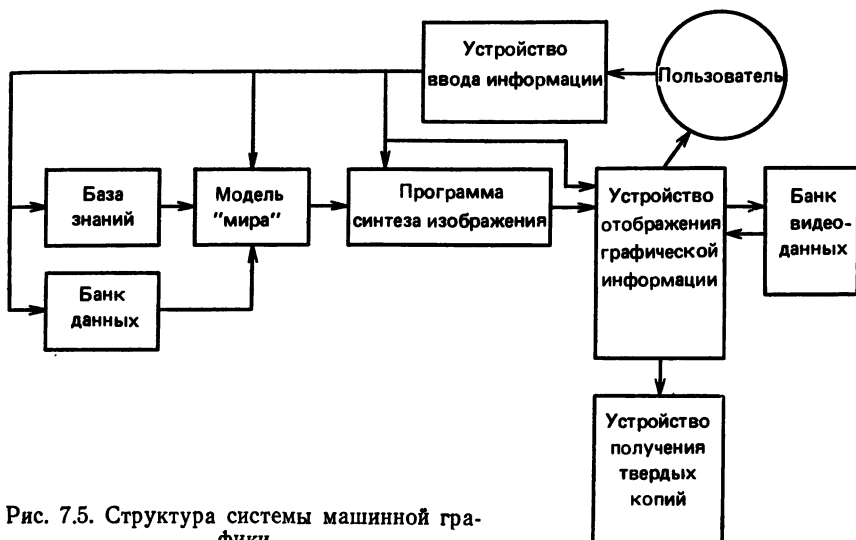


Рис. 7.5. Структура системы машинной графики

линдрическими или задаваться полиномиальными функциями. В моделях сплошных тел объекты формируются непосредственно путем комбинирования с помощью трехмерных теоретико-множественных операций (таких, как объединение, пересечение, разность) — так называемых примитивов. В качестве примитивов обычно используются простейшие трехмерные геометрические тела — параллелепипеды, пирамиды, цилиндры, сферы.

По способу описания объектов модели «мира» можно разделить на модели, задающие объект в виде набора данных и в виде операции его построения. Сама операция построения может быть представлена в процедурной форме или задаваться структурой данных. Способ описания, использующий полностью детерминированное задание всех исходных данных, характеризующих объект, основывается на концепции имитации процесса фотографирования реальной сцены. Это приводит к необходимости однозначного задания всех объектов модели «мира» в момент ее «фотографирования» воображаемым фотоаппаратом. Сейчас чаще используется задание объектов «мира» в виде операции их построения. В этом случае объект синтезируется непосредственно в процессе построения. Это значительно облегчает задачу разработчика модели, так как вместо детального описания объекта он задает лишь закон его построения (например, в виде закона распределения вероятности реализации конкретных фрагментов объекта). Кроме геометрического описания объектов в модели «мира» хранятся и другие параметры, необходимые для синтеза изображения (например, цвета поверхностей объектов).

При создании модели «мира» используются база знаний и банк данных. База знаний вводит фактически законы и ограничения, действующие при создании объектов «мира». Банк данных позволяет «запитувать» модель «мира» стандартными данными, необходимыми для ее работы. Банк данных и база знаний могут постоянно изменяться и дополняться пользователем в процессе эксплуатации системы.

Сформированную таким образом модель «мира» специальная программа преобразует в массив чисел, соответствующий изображению. Наиболее сложной задачей такого рода является синтез реалистичного изображения трехмерной сцены. Большой прогресс в этой области был достигнут к началу 80-х го-

дов. К настоящему времени в процессе синтеза цветного трехмерного реалистичного изображения можно выделить следующие этапы:

удаление скрытых поверхностей, т. е. поверхностей или их частей, которые не видны из точки нахождения воображаемого наблюдателя. Этот этап может быть выполнен, например, с помощью «алгоритма Z-буфера» или «алгоритма художника»;

закраска видимых поверхностей с учетом свойств самой поверхности (цвет, фактура, отражательная способность), ее относительного местоположения, ориентации, а также свойств источников света и других поверхностей;

выравнивание интенсивностей соседних поверхностей с помощью алгоритмов закрашки Гуро или Фонга [Ван Дам, 1986].

В последнее время благодаря быстрому удешевлению аппаратной части СМГ возрос интерес к синтезу динамических изображений в реальном масштабе времени, т. е. к так называемой компьютерной мультипликации. Динамическая графика находит применение в компьютерных играх, тренажерных системах, АСНИ. При синтезе динамических объектов, движущихся по законам модели «мира», программа синтеза изображения кроме перечисленных выше операций выполняет также ряд геометрических преобразований, соответствующих моделируемому движению.

Банк видеоданных служит для воспроизведения готовых изображений и их фрагментов. Он может постоянно пополняться новыми данными, полученными как извне, так и в процессе функционирования самой системы.

К устройствам отображения графической информации относятся векторные дисплеи, растровые дисплеи, дисплеи на запоминающих ЭЛТ, плазменные панели [Гардан и др., 1987; Гилой, 1981]. Наиболее популярны в настоящее время растровые дисплеи. В них изображение строится из набора дискретных элементов — пикселей. В современных растровых дисплеях может генерироваться изображение с дискретностью до 4096×4096 пикселей, с числом цветов до 4096 и числом уровней яркости одного цвета до 128.

К устройствам получения твердых копий изображения относятся графопостроители и принтеры. Обычно графопостроители используются для документирования полученных изображений, а принтеры, обеспечивающие более низкое качество воспроизведения, — для текущей работы.

В качестве устройства ввода информации в ЭВМ наиболее часто используются: алфавитно-цифровая и функциональная клавиатуры, световое перо, графический планшет, шар, манипулятор типа «мышь», а также различные устройства указания пальцем [Гардан и др., 1987]. Все эти устройства при вводе информации различного типа имеют свои преимущества и недостатки. Обычно пользователь имеет возможность выбора того устройства ввода, которое наиболее удобно для него в данный момент.

В качестве базовых для синтеза изображения могут использоваться универсальные ЭВМ практически всех классов — от персональных компьютеров до суперЭВМ. Простейшими графическими средствами снабжены и современные бытовые компьютеры. Для синтеза реальных трехмерных сцен необходимы 32-разрядные ЭВМ. Современная высокопроизводительная графическая система средней стоимости способна формировать динамическое изображение, состоящее из 1000 многоугольников с частотой 12 кадров в секунду [Ван Дам, 1986].

Разработка графических программ на языках высокого уровня началась в конце 60-х годов. При этом широко использовалась идеология «виртуальных» устройств, обеспечивающая независимость программного обеспечения высокого уровня от особенностей внешних устройств конкретной вычислительной системы. Программист пользовался лишь координатами виртуального экрана, а программы нижнего уровня автоматически переводили их в координаты реального экрана.

Приблизительно в этот же период времени были разработаны два пакета графических программ, доступные для всех типов дисплеев. Это трехмерная система Core Graphics System и двухмерная система Graphical Kernel System, принятая Международной организацией по стандартизации. Однако у этих па-

кетов есть существенный недостаток. Они разрабатывались еще до того, как стали отдавать предпочтение растровым дисплеям, поэтому при использовании этих пакетов невозможно в полной мере учитывать преимущества пикселей.

В настоящее время графические операторы (или вызовы графических подпрограмм) можно применять практически во всех языках высокого уровня (ПЛ/1, Фортран, Паскаль, Си, Бейсик и т. д.). Однако все еще не существует самостоятельного языка графического программирования, в котором основные элементы имели бы пиктографическую, а не текстовую природу.

В будущем благодаря быстрому уменьшению отношения цена/производительность следует ожидать появления программ, синтезирующих реалистичные динамические изображения на экранах персональных компьютеров. Что же касается мини- и суперЭВМ, то, по мнению ряда специалистов, уже к началу 90-х годов нынешнего столетия многие кадры фильма, целиком синтезированные ЭВМ, невозможно будет отличить от реально снятых кадров.

В перспективе ожидается переход к трехмерным дисплеям, основанным, например, на принципе голографического изображения. Отображаемая на экране информация будет лучше, чем в настоящее время, сопровождаться звуковыми эффектами. Процедура создания модели «мира» должна еще более упроститься, сведясь к заданию в удобной для человека форме наиболее общих законов и параметров, характерных для моделируемого «мира». Ведущую роль при этом должны сыграть идеи и методы искусственного интеллекта. В итоге экран дисплея должен стать фактически окном в созданный компьютером «мир», функционирующий по любым наперед заданным законам и активно отвечающий на любое воздействие человека.

Глава 8.

Состояние развития современных систем обработки визуальной информации

В. В. Александров, В. А. Анисимов, Н. Д. Горский, С. Н. Мысько

8.1. Автоматизированные системы обработки изображений

В соответствии с классификацией, введенной в § 7.1, существует три типа систем обработки визуальной информации: автоматизированные системы обработки изображений (§ 8.1); системы анализа изображений (§ 8.2); системы машинной графики (§ 8.3). Для каждого типа сначала приводится сводная таблица, содержащая общие характеристики основных систем этого типа (отечественных и зарубежных), а затем — описание наиболее интересных зарубежных систем. Описанию отечественных систем посвящена гл. 9.

Поскольку количество разработанных в настоящее время АСОИЗ очень велико, в табл. 8.1 представлены только те системы, которые отражают разнообразие направлений развития современных АСОИЗ. Так, ЦОГИ, СУТО-II, Context Vision относятся к крупным АСОИЗ с развитым аппаратно-программным обеспечением; SUN и СПЕКТР-Д — рабочие станции для создания проблемно-ориентированных систем; SCIA, BVS, ПРИЗНАК-1 — программно-аппаратные комплексы для решения конкретных задач; КРОКИС-2, ELIA — проблемно-ориентированные комплексы программ.

Ниже будут описаны следующие АСОИ: GSS [Rummel, 1986], Context Vision [Context, 1985], ELIA [Taniguchi et al., 1986], SUN-3/160 [Weston, et al., 1987].

Система GSS. Основное назначение этой проблемно-ориентированной системы — обнаружение и распознавание большого количества плоских деталей

Характеристики автоматизированных систем обработки изображений

Название (страна)	Назначение	Особенности реализации	Аппаратные средства	Источник
АСОВИ (СССР)	Классификация наземных объектов по материалам многозональных дистанционных данных, получаемых с аэрокосмических аппаратов	Подсистемы коррекции, улучшения качества и сегментации изображений работают со входными видеоданными. Подсистема классификации использует их результаты и априорные сведения, хранящиеся в банке картографической информации (легенды карт) и банке опорных данных	Реализация на ЕС ЭВМ (ОС ЕС)	Агаев и др., 1984
ГРАФИТ (СССР)	Ввод и распознавание в ЭВМ изображений, содержащих рукописную, символьную и графическую информацию	Базовый состав распознаваемых объектов: нестилизованное рукописное изображение русского и латинского алфавитов, цифры и специальные знаки. Пользователь имеет возможность дополнения состава распознаваемых объектов. В основу процедуры принятия решений заложены принципы использования визуального контекста и целенаправленного целостного распознавания (выдвижение и подтверждение	Ввод изображений в ЭВМ с устройств считывания графической информации ТС-6030 (рас-творная форма) и ЕС-7989 (векторная форма) в режимах on- и off-line. Реализация системы на ЕС ЭВМ (ОС ЕС, версия 6, 7)	Байков и др. 1986

КРОКИС-2 (СССР)	Обработка и анализ многозональных изображений (улучшение качества, сегментация, классификация)	типотез) Диалоговый пакет программ, содержащий 7 взаимосвязанных программных комплексов (4—обработка, 3—анализ). Результаты работы комплексов сегментации и улучшения качества служат входными данными для комплексов классификации, участков изображения, определения их статистических параметров и характеристик формы	Реализация на ЭВМ СМ-4 (РАФОС ОС РВ) и программно-совместимых с ней. Загрузка изображений через МД или МЛ; наличие устройства отображения не обязательно	Гиммельфарб и др., 1985
ОКРУС-4 (СССР)	Автоматическое распознавание символов и ввод в ЭВМ текста	Распознавание рукописных и машинописных знаков производится аппаратно-программным комплексом. Отделение знаков и выделение первичных признаков производится с помощью аппаратных блоков. Программно реализованы функции обобщения признаков и принятия решений	Фототелеграфный аппарат используется для ввода документа. Управление от ЭВМ «Электроника-60» (РАФОС). Обработка изображения в специальных блоках, работающих параллельно с ЭВМ. Скорость ввода информации 10—20 зн./с	Мадо, 1985
СТЗ (СССР)	Распознавание в сложной сцене трехмерных объектов известной формы на одиночном черно-белом телевизионном изображении	Исследовательский комплекс программ, моделирующий систему технического зрения. Моделируется работа матричного процессора с	Изображение вводится с промышленной телевизионной установки типа МТУ-16, работающей по принципу стробирования. Установка подключена к	Андреев, 1981; Андреев и др., 1987

Название (страна)	Назначение	Особенности реализации	Аппаратные средства	Источник
ПРИЗНАК-1 (СССР)	Автоматизированный экспресс-анализ изображений при геологической разведке в глубоководных районах океана	Цель анализа — обнаружение на дне железомарганцевых конкреций, оценка их количества и географических координат. Анализ проводится на бинаризованном изображении по набору заранее определенных признаков	ЭВМ типа СМ-4 (ОС РВ) через специальное устройство сопряжения и аппаратуру в стандарте КАМАК	Афанасьев и др., 1986
	Автоматизация исследований, построение проблемно-ориентированных систем обработки и анализа изображений	Рабочая станция ввода и обработки изображений с устройством отображения черно-белой (256 градаций) и цветной (256 цветов) видеоинформации (512×512 отсчетов). Имеет базовый набор программной обработки. Диалог — через меню функций — Программирование на языке Фортран	Дисплейный процессор на базе микропроцессорного набора серии K389 Шесть страниц памяти изображений. Интерфейсы связи с периферийным оборудованием и СМ ЭВМ	
СПЕКТР-Д (СССР)				Варченко и др., 1986

Обработка аэрокосмических изображений земной поверхности для исследований в области геологии, лесного хозяйства, гидрологии; автоматизация научных исследований	Библиотеки программ приема, хранения, обработки изображений объединенные по функциональным признакам в проблемно-ориентированные комплексы для решения различных прикладных задач	Базовые вычислительные комплексы БЭСМ-6, ЕС-1060 связаны с терминальными комплексами интерактивной обработки изображений на базе СМ-4 (ОС РВ) и дисплейных процессоров ОМЕГА, ОСКАР, МІР	Алексеев и др., 1983
Описание лиц по фото-портретам, ведение базы видеоданных, содержащей фотопортреты	В базе хранятся изображения (портреты), анкетные данные о людях и структурированные описания каждого изображения (СУБД АДАБАС). Распознавание лиц происходит по структурированным данным, совпадению с эталоном и выделенным видеопризнакам — характерным элементам лиц	Центральная ЭВМ- VAX-11/780 (VMS). Ввод изображений с телекамеры или сканирующего устройства. Хранение изображений в карусели микрофишей на 80 000 изображений	Brinkmann, 1982
Обработка и анализ графических, полутонных и цветных изображений для автоматизации исследований, биомедицинских приложений, криминалистики, обработки дистанционных данных	Интерпретированная система, поддерживающая функции хранения, предобработки, выделения признаков и классификации изображений. Имеется набор проблемно-ориентированных программных комплексов для решения прикладных задач (в частности — идентификации отпечатков пальцев, обнаружения объектов на земной поверхности и т. д.)	Многопроцессорная система с модульной архитектурой. Основные процессоры: управляющий (Motorola 68000), двусторонней фильтрации, плавающей арифметики, выборки данных, геометрических преобразований, бинарной обработки, управления дисплеями. ОС UNIX	Context, 1985

Название (страна)	Назначение	Особенности реализации	Аппаратные средства	Источник
СУТО-II (Институт окружающей среды, Мичиган, США)	Анализ изображений в цитологических, биологических и других медицинских исследованиях, распознавание объектов на изображениях	Широкий набор операций предварительной обработки изображений, выделения признаков, классификации объектов. Реализация быстрых операций поиска на изображении заданного видеосигнала	Специальный процессор обработки изображений конвейерного типа производительностью $1 \cdot 2 \cdot 10^8$ опер./с. Управляющая ЭВМ VAX 11/780 (VMS). Ввод с ТВ-камеры, электронного микроскопа, из сети ЭВМ	Lougheed et al., 1980
GSS (Сименс, Мюнхен, ФРГ)	Распознавание радио-технических и машинно-строительных деталей, которые могут касаться и перекрываться в поле зрения	Детали описываются набором признаков геометрическими эталонами, организованными в структурированную модель. Модели хранятся в памяти ЭВМ и используются при распознавании в режиме управления данными	Система содержит набор аппаратных модулей для выделения признаков; для управления и распознавания используются микропроцессоры INTEL 8086/8087. Хранение информации — на дисках. Ввод изображений — с телекамеры	Rummel, 1986
ELIA (Университет Киушу, Япония)	Обработка многообразных изображений, получаемых с американского спутника LANDSAT, с целью выделения и анализа территориальных ресурсов	Программный комплекс, в состав которого входят подсистемы архивации, преобработки и проблемно-ориентированного анализа изображений. Знания о предметной области представляются экспертными оценками оптимальных параметров	Реализация на оборудовании центра обработки изображений университета Киушу	Taniguchi et al., 1986

SCIA (Калифорнийский университет, Лос-Анджелес, США)	Обработка и анализ изображений с электронных микроскопа (распознавание элементов клетки, определение патологий)	Интерпретированная система, обеспечивающая ввод с ЭВМ, обработку, хранение и разнообразный анализ изображений. Базовое программное и аппаратное обеспечение управления 17 меню. Проблемно-ориентированные программы связаны с базовым ПО системой архивации изображений (DBASE-III) и пакетами статистической обработки (SAS)	обработки и правилами выделения и интерпретации объектов	Klinger, 1986
SUN Microsystems Inc., США)	Рабочая станция для проведения работ, связанных с подготовкой данных, обработкой и передачей в вычислительной сети текстовых и видеоданных	Набор операций для проведения предварительной обработки изображений (масштабирование, повороты, интерполяция, выделение границ), создания базы данных. Возможность подключения проблемно-ориентированных программных продуктов пользователя программ обработки изображений до 2048 × 2048 элементов, число одновременно воспроизводимых цветов до 256	Обработка и визуализация изображений на ЭВМ IBM PC/AT с адаптером EGA, монохромным полупроводниковым графическим дисплеем; хранение изображений на видеодиске (1 Гбайт) и двух винчестер-дисках по 65 Мбайт; вывод на лазерный принтер	Weston et al., 1987

и конструкционных элементов небольших размеров, находящихся в поле зрения. Детали могут иметь произвольную ориентацию, касаться друг друга и частично перекрываться. Ввод изображений осуществляется с телекамеры. Затем полутоновые изображения обрабатываются тремя спецблоками: выделения контуров, прослеживания линий и сегментации. Результаты обработки и исходные изображения поступают в процессор (Intel 8086/87), где производится идентификация деталей и вырабатываются сигналы об их положении и ориентации в поле зрения.

Процесс распознавания проходит в две стадии. На первой стадии с помощью спецблоков на полутоновом изображении выделяются примитивные образы и признаки типа углов, прямых линий, окружностей. Состав выделяемых признаков определяется моделями распознаваемых в данный момент деталей. На второй стадии осуществляется идентификация деталей. Для этого из выделенных примитивов синтезируются предполагаемые прототипы объектов, которые поочередно (для всех потенциальных деталей в поле зрения) сравниваются с существующими в памяти моделями деталей. Стратегия сравнения — последовательная по моделям согласно принципу «управления целью».

Большое значение для работы системы имеет построение моделей деталей, т. е. наполнение базы знаний. Модель детали состоит из геометрической модели, описывающей внешний вид детали в терминах геометрических примитивов и их пространственных взаимоотношений, а также из «генеративной» модели. Последняя содержит последовательность операций, которые используются для управления процессом идентификации детали, и список параметров, управляющих работой специализированных аппаратных блоков и программ обработки изображения. Геометрическая модель формируется в интерактивном режиме с использованием модулей обработки путем выделения на изображении детали примитивов с помощью светового пера. Процедурная модель, формируемая экспертом, представляет собой описания типичной последовательности операций, например, для идентификации фронтальной и задней поверхности детали.

Время распознавания зависит от сложности наблюдаемой сцены и сложности моделей деталей. Для сцен с 10—20 деталями оно составляет 1—4 с. Точность определения позиции деталей 0,5 элемента изображения (при растре 256×256 элементов), точность определения ориентации 1 град.

Система Context Vision. Данная система функционирует в Техническом университете г. Линчепинга (Швеция) с 1983 г. Она является хорошим примером развитой многопользовательской системы обработки и анализа изображений, работающей в режиме «множественный поток команд — множественный поток данных».

Основу системы составляет набор спецпроцессоров обработки и анализа изображений; процессор двумерных сверток, работающий в конвейерном режиме со скоростью 100 млн. опер./с; процессор плавающей арифметики (сложения, умножения, разложения в ряд Тэйлора) конвейерного типа, работающий со скоростью 14 млн. опер./с; процессор обмена с памятью, обеспечивающий передачу данных из памяти изображений в сверточный и арифметический процессоры; процессор геометрических преобразований, обеспечивающий операции изменения масштаба, поворота и другие линейные преобразования изображений; процессор побитовой обработки для булевых операций над бинарными изображениями, рекурсивных и нерекурсивных окрестностных операций; процессоры управления рабочими станциями (Motorola 68000), обеспечивающие обмен между ними, памятью изображений и устройствами ввода-вывода; процессор-супервизор (Motorola 68000), управляющий общей магистралью, к которой подключены все остальные процессоры, память изображений, магнитные диски (160 Мбайт), оптический диск, ТВ-камеры, принтеры и плоттеры. Типичная конфигурация системы имеет три рабочие станции, каждая из которых ориентирована на решение своего круга задач. Проблемная ориентация обеспечивается развитыми программными средствами, которые включают комплексы общей обработки изображений, кодирования и сжатия видеоданных, обработки аэрокосмической информации, идентификации отпечатков пальцев, обработки

изображений с микроскопов. Взаимодействие с пользователем происходит через иерархические меню возможных функций. Допускается разработка специализированных программ на Си, Фортран-77, Паскале в среде ОС UNIX. В первой версии системы не было предусмотрено использование знаний для решения задач анализа изображений, система имеет транслятор с языка ЛИСП и ряд инструментальных средств для ведения базы знаний.

Система ELTA. Эта система является примером проблемно-ориентированного программного комплекса для анализа мультиспектральных изображений, получаемых с американского спутника LANDSAT. Назначением системы является извлечение из изображений информации о географических объектах (преимущественно водных) в интересах землепользования. В состав системы входят блоки сегментации изображений, ведения архива видеоданных, предобработки изображений, ведения базы знаний, управления диалогом.

Процесс выделения информации протекает в два этапа. На первом этапе производится выделение водных объектов на основе преобразования координат цвета элементов изображения в обобщенные яркость, фазу и «контрастность» спектра и анализ определенных зон каждой из этих составляющих, затем — отделение рек от морей и озер на основе геометрических признаков выделенных областей и правил отнесения отдельных участков к одному из классов. На втором этапе осуществляется прослеживание рек от устья к истоку и от истока к устьям на основе правил, описывающих структуру речной сети и применение детекторов-масок. В заключение проводится интерпретация полученных данных: определяются длины рек, координаты их истоков, устьев, мест впадения притоков, параметры береговых линий и т. д. Система реализована на языке Си и работает в центре обработки изображений университета Киуши, Япония.

Рабочая станция SUN-3/160. Рабочие станции становятся мощным средством для разработки прототипов и эксплуатации действующих систем анализа изображений, автоматизации проектирования. Одной из наиболее удачных реализаций рабочих станций является SUN-3/160.

Эта станция в качестве центрального процессора и математического сопроцессора использует соответственно Motorola 68020 и Motorola 6881, работающие на частоте 16,67 МГц. Производительность станции около 2 млн. опер./с. Оперативная память включает до 16 Мбайт, виртуальная 256 Мбайт, графический процессор поддерживает отображение 256 цветов из 16 млн. возможных на экране с разрешением 1152 на 900 точек. Жесткие диски могут быть емкостью от 71 Мбайта до 1,1 Гбайта при среднем времени доступа около 20 мс. Станция включает также дисковод с гибкими дисками (360 Кбайт и 1,2 Мбайт), устройство записи на магнитную ленту (кассету, 60 Мбайт).

Операционная система SUN OS является вариантом операционной системы Berkley 4.2 BSD (подобной UNIX), включающей возможности систем System V. SUN OS поддерживает работу в сети (Network File System) при одновременном доступе для нескольких пользователей, каждый из которых может использовать вплоть до 256 Мбайт виртуальной памяти. Станция обеспечивает также работу в сети Ethernet. Интерфейс пользователя улучшен за счет использования трехклавишного манипулятора «мышь», графических и текстовых окон на экране, представляющих меню, демонстрирующих результаты и состояние выполняемых программ. Важным моментом является наличие сопроцессора Sun IPC (10 МГц Intel 80286 и 8 МГц Intel 80287), 1 Мбайта оперативной памяти, графической карты и других средств, позволяющих выполнять программы, написанные для IBM PC/AT и совместимых персональных ЭВМ в операционной системе MS DOS 3.1.

Указанные технические мощности станции предоставляют пользователю широкие возможности разработки систем анализа изображений (полутонных, цветных, графических). В режиме многооконного интерфейса и управления с помощью «мыши» через экранное меню легко выполнять операции предварительной обработки: масштабирование, выделение контуров, модификацию гистограмм, повороты изображений на любой заданный угол, выделение участ-

ков изображения произвольной формы и их модификацию и др. Различные результаты обработки могут отображаться в различных участках экрана (окнах) одновременно, записываться в архив (файловую систему либо базу данных). Наличие широко распространенных компиляторов с таких языков, как Паскаль, Си, Фортран, позволяет пользователю создавать проблемно-ориентированное программное обеспечение, подключать свои библиотеки к уже существующим в системе. На базе языков ЛИСП. Пролог, объектно-ориентированных языков и интегрированных программных средств (OPS83, LOOPS) становится возможным проектировать базы знаний и объединять их с модулями обработки изображений для построения экспертных систем анализа видеоданных.

8.2. Системы анализа изображений

В настоящее время существуют разные подходы к построению систем анализа изображений. В соответствии с основной тематикой данной книги будут рассмотрены интеллектуальные САИ, называемые экспертными САИ (ЭСАИ), ЭСАИ объединяют САИ с технологией баз знаний (см. гл. 13). В связи с этим в табл. 8.2 выделены параметры САИ, характеризующие как анализ изображений, так и технологию использования знаний.

В этом параграфе будут описаны следующие ЭСАИ: 1) ACRONYM; 2) система, разработанная Nagao и Matsuyama; 3) система Niyogi и Srihari; 4) SIGMA. В связи с тем, что SIGMA оказалась одной из наиболее успешных реализаций системы анализа изображений, описание этой системы дается наиболее полно.

Система ACRONYM. Эта система предназначена для решения задачи распознавания и локализации объектов на фотоснимках. Система использует геометрические модели реальных трехмерных объектов и формализованные знания экспертов. По мнению [Binford, 1982], эта САИ наиболее близка к универсальной системе машинного зрения по принятой методологии ее построения. Ядро системы составляют геометрические модели класса объектов (пассажирские авиалайнеры компаний Lockheed и Boeing); правила, определяющие операции проективной геометрии при отображении трехмерных объектов на двумерное изображение для различных точек наблюдения, а также алгоритмы выделения границ, анализа текстуры, формы и др.

Система анализирует изображения на основе стратегии «управление целью». ACRONYM начинает работу с анализа моделей (трехмерных) и выделяет все поверхности и контурные линии, которые могут быть видны на снимке и являются наиболее важными для идентификации объекта. Полученные данные преобразуются с учетом расположения камеры, куда поступает реальное изображение сцены, при этом производятся все необходимые операции: повороты, сдвиги, коррекция описаний форм и т. п. Затем выполняется проектирование полученной трехмерной конструкции объекта на плоскость двумерного изображения. ACRONYM строит граф предсказания, в узлы которого помещаются характеристики форм, образованных контурными линиями, а дуги описывают соотношения между ними. Система определяет участки изображения в терминах контурных полос (ribbons) и эллипсов и выполняет локальные сопоставления конфигураций полос и эллипсов, наблюдаемых на изображении с ожидаемыми на основе графа предсказания. Таким образом, локализуются участки изображения с вероятным расположением, например, крыла самолета или фюзеляжа. Далее проверяется состоятельность глобальных интерпретаций на основе графа, отражающего соотношения часть-целое. При этом должны выполняться отношения, описываемые дугами графа предсказаний; получаемые ограничения в результате интерпретации каких-либо участков изображения как частей изображения не должны противоречить геометрической модели.

Достоинством системы является возможность оперирования классами объектов. Конкретные марки самолетов представлены в модели как ограничения

на данный класс объектов; применение трехмерных моделей позволяет добиться относительной инвариантности по отношению к точке наблюдения сцены. Недостатком является слабое использование знаний экспертов о возможном окружении объектов на аэрофотоснимках, что, по-видимому, является одной из основных причин ошибок распознавания, допускаемых системой на снимках с большой детальностью и многими объектами (более трех самолетов). Согласно [Binford, 1982] недостатком также является то, что интерпретация производится на уровне изображений, а не трехмерных тел; сегментация плохо учитывает текстурные параметры; система не способна локализовать полосы небольших размеров; одной стратегии (управление целью) недостаточно, и необходимо использовать также стратегию «управление данными».

Система Nagao, Matsuyama. Система предназначена для выполнения содержательной интерпретации аэрофотоснимков местности и способна «узнавать» объекты следующих классов: поля с растительностью или без нее, лесные массивы, реки, дороги, здания, урбанизированные территории. В базе знаний хранятся свойства перечисленных объектов, отражающие такие характеристики, как яркость, текстурные признаки, протяженность, параметры остова объекта, контрастность и др. На различных этапах анализа изображений система работает согласно стратегиям управления данными либо управления целью. Процесс анализа условно можно разбить на пять этапов: предварительная обработка; сегментация на области, однородные по яркости (в различных спектрах); выделение особых областей (cue regions); анализ каждой особой области в соответствии с ее типом; анализ характеристик особых областей и принятие решения (окончательная интерпретация, повторная сегментация и др.). Особые области разделяются на однородные по яркости, протяженные, теневые (образующие тени от объектов), области с растительностью, поверхности водоемов, области с высококонтрастной текстурой.

На начальном этапе система следует стратегии управления данными: производится сглаживание с целью устранения шума, а также консервативная сегментация путем наращивания областей с однородными по яркости свойствами. Далее определяются особые области. Протяженные области определяются по прямоугольнику и по «остову» объекта, теневые — по гистограмме яркости, области с растительностью, поля, лесные массивы — по текстурным признакам (количеству перепадов яркости в окне определенного размера), а также по соотношению яркостных характеристик в различных спектрах. Для каждой особой области измеряются свои параметры (прямолинейность границ участков полей, протяженность рек, дорог и др.). Получив начальную карту (план) сегментации, система переходит к стратегии управления целью: получению интерпретации областей на основе формализованных знаний о свойствах объектов и их соотношениях. Так, здания должны находиться на урбанизированной территории, иметь прямые углы, возможны теневые участки, прилегающие к зданию в соответствии с расположением солнца в момент съемки местности; высококонтрастные области, граничащие с теневыми, интерпретируются как лесные массивы, отсутствие теней позволяет рассматривать области как участки с травяной растительностью (с учетом цветовой окраски) и т. д.

В конфликтных ситуациях система инициализирует операции расщепления-слияния областей (например, разделение области в узком месте и объединение получившихся частей с прилегающими областями).

Основными недостатками системы являются: использование характеристик объектов, специфичных для узкого круга задач и эвристических правил интерпретации; слабое использование методов описания форм объектов, текстур (фактически система просто разделяет области с текстурой и без нее), контекста.

Система Niyogi, Srihari. Разработки [Niyogi et al., 1986; Wang et al., 1986] представляют собой широко используемый подход к построению ЭС, применяемый также при создании ЭСАИ. Этот подход заключается в использовании системы правил, с помощью которых представляются знания в ЭС и произ-

Характеристики систем

Название (страна)	Анализируемые изображения	Назначение	Сегментация
ACRO- NUM (США)	Полутоновые цветные изображения с аэро- фотоснимков	Распознавание и локали- зация объектов самоле- тов определенного типа на местности	Многоступенчатая; выделение контурного рисунка и его сегмен- тация на полосы и эллипсы
LandScan (США)	Стереоизображения по аэрофотоснимкам урбанизированных территорий	Содержательная интер- претация аэрофотосним- ков, указываемых в ин- терактивном режиме ра- боты с системой	Выделение контурных линий и восстановле- ние трехмерных по- верхностей
(LS81) (Канада)	Полутоновые цветные изображения уличной сцены; 128×128 эле- ментов	Содержательная интер- претация изображений уличной сцены	Многоступенчатая; слияние-расщепление областей, однородных по цвету и яркости
MAP- SEE-2, MAP- SEE-3, (Канада)	Стилизованные кон- турные рисунки карт местности	Содержательная интер- претация контурных карт, сопоставление кар- ты с изображением сце- ны	Выделение сегментов контурных линий, их объединение в цепоч- ки и определение по ним областей
MSIAS (США)	Изображения Земли в 7 спектрах, полу- ченных со спутника	Классификация объектов на многоспектральных аэрофотоснимках	Сегментация на осно- ве гистограмм ярко- стей в различных спектрах по прави- лам, определяемым пользователем в ин- терактивном режиме
(NM80 (Япония)	Многоспектральные аэрофотоснимки, каж- дый 256×256 элемен- тов, 8 бит/эл.	Содержательная интер- претация аэрофотосним- ков местности	Многоступенчатая сегментация по ярко- сти, текстуре в раз- личных спектрах

анализа изображений

Представление знаний, модели, стратегия	Особенности реализации, тип ЭВМ, языки программирования	Источник
Геометрическое описание классов трехмерных объектов (самолеты типа Lockheed, Boeing); правила операций проективной геометрии; отношения, связывающие компоненты объекта. Используется стратегия управления целью		Brooks, 1981; 1983
Список объектов и матрица отношений между ними. Используется стратегия управления целью и данными	LISP. Анализ запроса пользователя на естественном языке, формирование цели	Bajacsy et al., 1985
Реляционная база данных, описывающая свойства изображений объектов, а также отношения между объектами. Используются стратегии управления целью и данными	PDP-15	Levine et al., 1981
Иерархия абстракций в виде графа обобщения-специализации; сеть фреймовых структур, представляющих отношения между объектами, а также их свойства. Используются стратегии управления целью, данными	MAPSEE-3. Отличается от MAPSEE-2 графом иерархии абстракций	Havens et al. 1983; Mulder, 1985
Система правил-продукций, описывающих свойства объектов и способы их классификации, а также иерархию абстракций (вспаханное поле, песок, глина являются почвой, почва и вода — часть сцены). Используется стратегия управления целью	SYMBOLICS 3600, VAX 11/780; ZETALISP, FLAVORS. Меню-интерфейс с пользователем, позволяющий выбирать и редактировать спектральные изображения, модифицировать базу знаний	Ferrante et al., 1984
Модель, представляющая ограничения на свойства (цвет, текстура, яркость, протяженность и т. д.) классов объектов (поля с растительностью и без нее, здания, леса, дороги, реки). Используются стратегии управления, целью, данными	Время преобработки 240 с. время интерпретации 200 с	Nagao et al., 1980

Название (страна)	Анализируемые изображения	Назначение	Сегментация
(NS86) (США)	Цветные полутоновые изображения доку- ментов, почтовой кор- респонденции	Содержательная интер- претация изображений, документов, корреспонден- ции, научных статей	Динамическая в про- цессе интерпретации по яркости, цвету, текстуре
(RB84) (США)	Полутоновые изобра- жения с аэрофото- снимков урбанизиро- ванной территории	Распознавание и локали- зация объектов на аэро- фотоснимках городских кварталов	Выделение областей, однородных по ярко- сти на основе алго- ритмов слияния-рас- щепления
SCERPO (США)	Полутоновые изобра- жения сцены из пред- метов в помещениях 512×512 эл.	Содержательная интер- претация изображений сцен внутри помещений	Выделение и связыва- ние линейных участ- ков контурных линий
SIGMA (Япония)	Аэрофотоснимки ур- банизированных тер- риторий	Содержательная интер- претация аэрофотосним- ков	Динамическая в про- цессе интерпретации по запросу управля- ющих модулей систе- мы. Различные пара- метры сегментации для различных клас- сов объектов
SPAM, MAPS (США)	Полутоновые цветные изображения с аэро- фотоснимков аэропор- та, снимков местности	Содержательная интер- претация изображений аэропорта, городских кварталов, местности, ин- формационный поиск в базе данных и изобра- жений	Двухступенчатая; слияние-расщепление областей, однородных по яркости, текстуре, выделение контуров
VISIONS (США)	Полутоновые цветные изображения уличной сцены 512×512 эл.; 6 бит/эл.	Содержательная интер- претация уличной сцены	Иерархическая по уровням пирамидаль- ной структуры дан- ных. Выделение обла- стей и контуров

Представление зданий, модели, стратегия	Особенности реализации, тип ЭВМ, языки программирования	Источник
Система правил на основе предикатов. Используется стратегия управления целью	VAX 11/780; PROLOG, LISP, С. Время анализа изображения почтового конверта около 1 с.	Niyogi et al. 1986
Система правил-продукций, описывающих свойства объектов, методы формирования контекста, гипотез, выбора стратегии и т. д., а также иерархию абстракций. Используются стратегии управления целью, данными	LISP Обработка и анализ изображений производится на пирамидальной структуре данных	Rosenthal et al., 1984
Модель «проволочного каркаса» трехмерных объектов. Используется стратегия управления данными.	VAX 11/750; VICOM-спецпроцессор; FRANZ-LISP, С. Время ввода и перестройки 30 с, время сегментации 40 с	Lowe, 1985
Сеть фреймовых структур для представления свойств объектов и отношений между ними, способов сегментации. Используются стратегии управления целью и данными	VAX 11/780, FLAVOR, LISP, C	Matsuyama et al. 1985
Фреймовые структуры для описания отношений между объектами, система правил, определяющая операции обработки изображений, генерацию гипотез и их проверку. Используются стратегии управления целью и данными	Включает базу видео- и текстовых данных	McKeown et al. 1983; McKeown, 1982
Реляционная база данных, описывающая граф соотношения областей, сегментов линий, вершин для классов реальных объектов. 11 модулей базы знаний («источников знаний»), определяющих операции обработки изображений, правила интерпретации. Используются стратегии управления целью, данными	PDP-15, CYBER 74; LISP. Обработка и анализ изображений производится на пирамидальной структуре данных	Hanson et al. 1978

Название (страна)	Анализируемые изображения	Назначение	Сегментация
(Y86) (канада)	Рентгеновские снимки череп 256×256 эл.; 8 бит/эл.	Распознавание особых участков на рентгенов- ском снимке	Одноступенчатая; вы- деление контурных линий на всем изо- бражении

водятся необходимые для логических выводов операции формализованными знаниями.

Функциональное назначение рассматриваемой ЭСАИ — анализ и содержательная интерпретация изображений печатных материалов (прежде всего — корреспонденции) для автоматизации процесса сортировки почтовых отправлений. Основная задача системы состоит в правильном обнаружении на почтовом отправлении адресов получателя и отправителя, после чего адрес получателя может быть проанализирован оптическим распознающим устройством. Если для почтовых писем и открыток число вариантов месторасположений текста с адресами корреспондентов ограничено, то для корреспонденции такого рода, как журналы, рекламные проспекты, этикетка с адресом может находиться в любом месте на обложке корреспонденции. Поэтому система должна быть способной отличать иллюстрационный материал, графические изображения, текст [Wang et al., 1986]. Другим примером использования системы может служить решение задачи анализа изображений научных статей, опубликованных в журналах [Niyogi et al., 1986].

Рассматриваемая ЭСАИ состоит из подсистемы обработки изображений, базы знаний и управляющей структуры. Изображения печатных материалов подвергаются предварительной обработке: фильтрации с целью устранения шума, после чего выполняется начальная сегментация, задача которой — разбить изображение на однородные области, из которых на последующих этапах анализа должны быть получены «логические блоки» (заглавие статьи, фамилии авторов, основной текст, рисунки, сноски и т. п.).

Значения в системе предоставляются правилами, которые описывают характеристики «логических блоков» — компонент страницы печатного материала, а также отношения между ними. Система поддерживает только пространственные соотношения. Правила базы знаний определяют также операции обработки изображений, которые возможно применять для различных типов «логических блоков».

Управляющая структура системы состоит из правил двух типов: управляющих и стратегических. Первые выполняют интерпретацию получаемых промежуточных результатов, инициализируют выполнение правил базы знаний, решают, на каких «логических блоках» следует акцентировать внимание. Например, управляющие правила на некотором этапе анализа могут решить, что в первую очередь нужно проводить анализ «сносок-примечаний», инициализируя таким образом выполнение только тех правил, которые имеют отношение к данному типу «логических блоков». Стратегические правила оценивают состоятельность получаемой интерпретации и определяют последовательность инициализируемых управляющих правил. Для оценки состоятельности получаемой интерпретации каждому правилу базы знаний приписывается значение достоверности от 0 до 1. Суммарное значение, получаемое на основании достоверностей

Представление знаний, модели, стратегия	Особенности реализации, тип ЭВМ, языки программирования	Источник
Сеть фреймов, описывающих ориентацию, протяженность особых участков рентгенограмм, а также пространственные отношения между участками. Используется стратегия управления данными	VAX-11/780. Время полного анализа 3 мин	Yan et al., 1986

всех правил, работавших при интерпретации, дает оценку состоятельности интерпретации, анализируемой стратегическими правилами.

Общая стратегическая линия работы системы соответствует подходу управления целью. Поэтому сегментация изображений производится динамично в процессе интерпретации при поступлении запроса от управляющих правил. Большая часть правил системы реализуется на языке, в котором стратегия управления целью осуществляется обратным прослеживанием по цепочке правил от результата к условию его получения.

Испытания системы проводились с почтовой корреспонденцией, как стандартной (анализируемой оптическими читающими автоматами), так и нестандартной. Более 80 % корреспонденции система проанализировала без ошибок, затрачивая на 1 страницу текста около 1 с. Работы по совершенствованию системы продолжают.

К недостаткам системы следует отнести прежде всего сложность дополнения новыми правилами, а также эвристический путь определения достоверности для правил базы знаний, что усложняет модификацию базы знаний и может привести к противоречивости используемых правил.

Система SIGMA. Данная система может рассматриваться как прототип системы анализа изображений с различной проблемной ориентацией. Поэтому целесообразно привести более подробное ее описание.

Функциональное назначение системы состоит в содержательной интерпретации аэрофотоснимков. Эксперименты проводились с полутонными изображениями урбанизированных территорий. SIGMA предполагает три функциональных уровня, соответствующих в системе трем различным экспертизам (рис. 8.1). Уровень эксперта GRE является основным модулем, управляющим всей системой. GRE собирает факты (результаты интерпретации), проводит их анализ и принимает решения: о выборе стратегии (управление целью или данными), об инициализации работы эксперта MSE, о состоятельности полученной интерпретации.

Эксперт выбора модели MSE использует информацию о контексте, полученную от эксперта GRE, для определения предполагаемого внешнего вида объекта, который необходимо локализовать на снимке. Например, эксперт GRE поручает эксперту MSE определить внешний вид объекта класса «здание» в соответствии с контекстом. MSE решает, что объект может предстать перед наблюдателем в виде прямоугольника с определенными ограничениями. Для этого MSE выполняет соответствующие преобразования для проекции трехмерного объекта на двумерное изображение.

Внешний вид объекта, определенный экспертом MSE, является исходной информацией для эксперта нижнего уровня машинного зрения LIVE, который выбирает соответствующие операторы обработки изображений, определяет последовательность обработки на основе формализованных знаний о методах обработки.

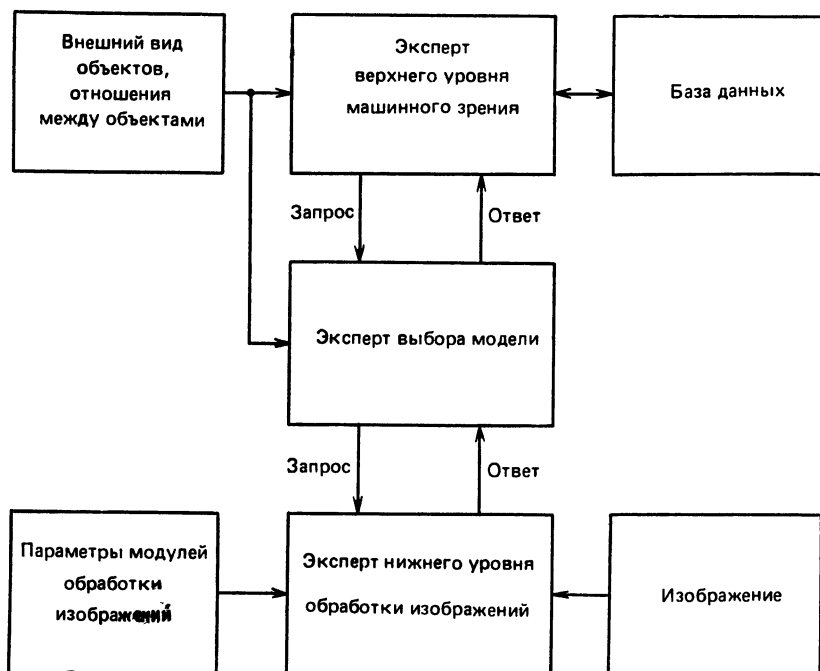


Рис. 8.1. Схема системы SIGMA

Формализованные знания в системе представляются сетью фреймов, описывающих объекты и отношения между ними. Характеристики объекта, а также отношения с другими объектами хранятся в слотах фреймов в виде продукций (правил). Левая часть продукции — условие выполнения действий, указанных в правой части, содержит собственно характеристики и отношения между объектами. В случае истинности условия в левой части продукции выполняется последовательность действий, в результате которой генерируется или устанавливается истинность гипотезы (например, о присутствии объекта «подъездной путь» рядом с объектом «здание» или о нахождении объекта «здание» на некотором участке изображения). В системе используются следующие типы отношений между объектами (и соответственно фреймами): 1) отношение типа часть-целое PW, представляющее объекты со сложной внутренней структурой; 2) пространственные (геометрические) отношения SP между объектами; 3) отношения типа специализация-обобщение АКО; 4) примеры из класса объектов IO; 5) конфликтные отношения (невозможность нахождения различных объектов на одном и том же участке изображения) ICW; 6) отношение «внешний вид объекта» APO. На рис. 8.2 представлен пример структуры знаний, используемых в системе.

Рабочим полем (областью общей памяти) системы является база данных. Все результаты промежуточных интерпретаций, получаемых во время анализа изображений, записываются в эту базу данных. Результатами промежуточных интерпретаций являются конкретные примеры объектов, их предполагаемое расположение, свойства и ограничения, связанные с конкретными объектами. Пространственное расположение как уже локализованных объектов, так и предполагаемых описывается двухмерным массивом (планом), в котором обозначаются области соответственно результатам текущей интерпретации.

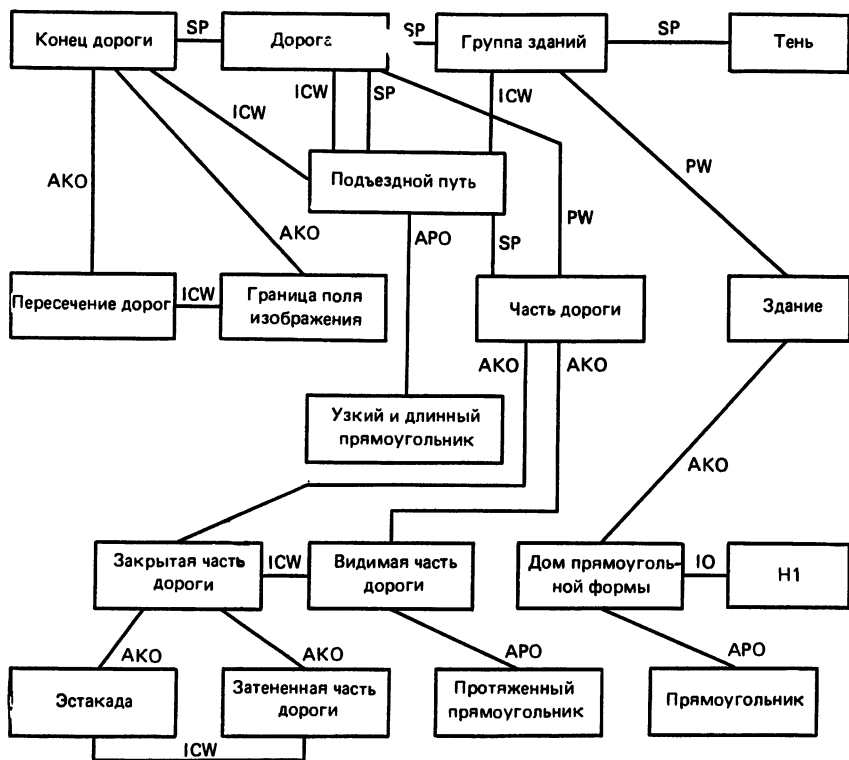


Рис. 8.2. Система отношений между объектами (фреймами)

Области могут перекрываться. Это означает, что существует несколько гипотез о нахождении различных объектов в одном и том же месте либо получены данные о нахождении в одном месте различных объектов. Перекрытие областей считается «состоятельным» (consistent), если ограничения, связанные с соответствующими объектами, не противоречат друг другу. Например, объекты класса «здание» и «группа зданий» могут привести к «состоятельному» перекрытию областей в плане. Во время работы системы база данных постоянно пополняется новыми фактами, изменяется план, проверяются вновь полученные соотношения между конкретными объектами, генерируются новые гипотезы.

Приведем один из возможных путей генерации гипотез на основе отношения часть-целое PW. Пусть объекты классов 01, 02, 03 связаны отношением PW (рис. 8.3,а). Пусть также на основе определенных признаков на изображении установлен факт наличия объекта s класса 02. Тогда в соответствии с существующим отношением PW предполагается существование объекта p класса 01 (с ограничениями, обусловленными объектом s). Такая «ситуация» порождает гипотезу о наличии объекта класса 03, связанного с объектом p отношением PW. Описание предполагаемого объекта класса 03 будет получено с помощью функции f , хранимой во фрейме описания объекта класса 01. После этого система попытается обнаружить объект класса 03 на изображении с учетом ограничений, сформированных в процессе порождения гипотезы.

Процесс анализа изображения инициализируется экспертом MSE, который на основе формализованных знаний выбирает объекты, наиболее простые для

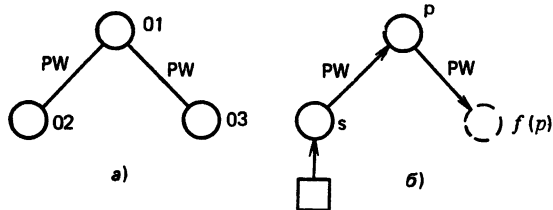


Рис. 8.3. Генерация гипотез на основе отношений часть-целое:

α — соотношения объектов классов 01, 02, 03; β — генерация гипотезы об объекте класса 03 на основе фактов присутствия объекта класса 02

обнаружения, и формирует задание эксперту LIVE для поиска соответствующих признаков-примитивов на изображении. Результаты работы записываются соответственно отыскиваемым объектам, помещаются в базу данных экспертом MSE и служат исходным материалом для работы эксперта GRE. Таким образом, в отличие от других систем анализа изображений, где начальная сегментация проводится независимо от контекста путем наращивания областей, однородных по яркости или текстуре, SIGMA выполняет целенаправленную сегментацию (в соответствии с запросом эксперта MSE).

Получив результаты начальной сегментации, эксперт GRE выполняет итеративный процесс интерпретации. Процесс заканчивается, когда не будет изменений в базе данных и будут соблюдены следующие условия.

1. Каждый факт обнаружения конкретного объекта приводит к генерации гипотезы о наличии других объектов в соответствии с определенными для данного класса объектов отношениями с помощью продукций, хранимых во фрейме описания объекта.

2. Все результаты интерпретации (гипотезы, факты обнаружения объектов) записываются в базу данных.

3. «Состоятельные» результаты образуют «ситуации».

4. Поскольку «ситуаций» обычно много, выбираются только наиболее достоверные. Чтобы определить достоверность, каждому факту и гипотезе о наличии объекта приписываются значения достоверности, на основе которых определяется достоверность результатов интерпретации.

5. Выбранные «ситуации» разрешаются проверкой предсказаний системы на основе ранее обнаруженных конкретных объектов либо путем обнаружения недостающих объектов на основе стратегии управления целью.

Завершающим этапом работы системы является выбор наиболее состоятельной интерпретации. Эта операция выполняется экспертом GRE путем отбраковки результатов интерпретации, не получивших достаточной достоверности.

Работа по совершенствованию системы SIGMA продолжается. Согласно [Matsuyama et al., 1985] слабыми местами системы являются: а) способ установления достоверности результатов интерпретации (искажения, шум на изображениях могут порождать ложные источники фактов, учитываемых в процессе интерпретации с возможно высокими значениями достоверностей); б) последовательный характер глобального управления (операции, выполняемые экспертом GRE) плохо согласуется с возможностью параллельной работы других подсистем. Авторы считают, что необходим более мощный механизм глобального управления работой всей системы.

8.3. Системы и пакеты машинной графики

В табл. 8.3. приведены основные характеристики систем и пакетов машинной графики. Более подробно дается описание пакетов ГРИФ и SPACE, систем GRAFIX и PIXAR.

Пакет прикладных программ (ППП) ГРИФ. Пакет ГРИФ является одним из самых развитых отечественных пакетов диалогового проектирования печатных плат. Он предназначен для диалоговой доработки эскизов трассировки пе-

чатных плат, выполненных конструктором вручную или полученных в результате выполнения автоматической трассировки на других ЭВМ, а также для автоматизации конструкторской документации. В качестве базовой используется ЭВМ типа СМ-4.

Пакет ГРИФ оперирует графическими данными на языке ЯГТИ. Описание законченного изображения представляется так называемым листом ЯГТИ. Лист включает паспорт, содержащий имя листа, габаритные размеры печатной платы, шаг координатной сетки. Каждый лист состоит из последовательности разделов, которые представляют собой совокупности абзацев, содержащих описание графических элементов одного типа. В абзаце имеются паспорт и графические данные. Паспорт включает информацию о типе графических данных и визуальных свойствах объектов, описываемых в абзаце. ЯГТИ позволяет задавать такие элементы, как ломаные линии, дуги, стандартные графические элементы, тексты.

Для осуществления графического диалога в системе формируется смешанный дисплейный файл, в который помимо геометрических данных включаются данные на языке ЯГТИ. В состав базы данных входит библиотека групповых элементов. Описание группового элемента состоит из разделов, в которых задаются различные геометрические представления элемента, используемые при решении разных задач проектирования. Таким образом, в ППП ГРИФ данные о проектируемой печатной плате описываются в основном в терминах машинной графики. Благодаря такой организации данных обеспечивается высокая реактивность системы в интерактивном режиме работы.

Пакет ГРИФ имеет развитый язык графического диалога, позволяющий задавать сложные преобразования графических объектов. При графическом диалоге возможно использование окна, для которого строится отдельный дисплейный файл. Все коррекции в этом случае выполняются как в файле окна, так и в основном файле, что обеспечивает быстрое формирование изображения в окне и учет результатов редактирования.

Пакет прикладных программ SPACE. Пакет предназначен для поддержки трехмерных графических задач САПР и может эксплуатироваться на БЭСМ-6, ЕС ЭВМ, СМ ЭВМ. Объем пакета — около 10 000 операторов языка Фортран.

О помощью ППП SPACE реализуются следующие графические средства:

- описание и хранение геометрических форм с использованием линейного представления или параметрического задания кривых и поверхностей по методу Кунса;

- конструирование объектов с помощью аффинных преобразований, теоретико-множественных операций, вращения и сдвига;

- построение проекций объектов и их вывод на графическое устройство.

Пользователь может описывать объекты, используя либо линейное (многоугольники и многогранники), либо параметрическое (кривые и поверхности) представление. Основной структурной единицей описания объекта в программах пакета служит вещественный массив. Для хранения описаний объектов предназначен специальный оперативный архив. Средства архива позволяют идентифицировать объект при его формировании программной переменной и в дальнейшем применять ее при работе с объектом.

Предусмотрены следующие способы формирования объектов:

- параметрическое задание объектов, т. е. определение объекта некоторой совокупностью параметров, которые полностью определяют его форму и положение в пространстве;

- описание объекта путем задания набора его составных или граничных элементов, описанных ранее и возможно модифицированных с помощью аффинных преобразований;

- построение объекта, являющегося результатом теоретико-множественной операции над двумя ранее описанными объектами (объединение, пересечение, разность);

Характеристики систем и пакетов машинной графики

Наименование системы (страна)	Назначение	Особенности реализации	Аппаратные и программные средства	Источник
ГРАН (СССР)	Автоматизация графического оформления результатов НИР и ОКР	Пользователь формулирует задачу на проблемно-ориентированном языке. При изменении формы представления результата возможна быстрая перестройка системы пользователем, не знакомым с языками программирования	СМ ЭВМ	Горин, 1984
ГРИФ (СССР)	Доработка эскизов трафаретных печатных плат	При решении различных задач проектирования используются различные геометрические представления одного и того же элемента. Данные о проектируемой печатной плате описываются в основном в терминах МГ	СМ-4, ЯГТИ	Ойхман, 1986
ДСТЧ (СССР)	Исследование проблем теории чисел на основе методов когнитивной компьютерной графики	Свойства исследуемых математических объектов отображаются на экране в виде пространственных и цветовых отношений элементов изображения. Используется декларативные знания предметной области (теории чисел), математические модели отдельных задач, प्रकारа связи абстрактных и графических объектов. Имеется обучающая подсистема с архивом «мультфильмов», иллю-	СМ-1420, СМ-1600; ОС РАФОС; IBM PC; Фортран Бейсик	Зенкин, 1983; 1987

ИНТЭАР (СССР)	Графическое моделирование трехмерных объектов на перспективных и аксонометрических изображениях, а также в ортогональных проекциях	стрирующих с объяснениями примеры решения задач	Предусмотрена возможность моделировать поверхность на основе ее линии очертания, нарисованной пользователем на экране графического дисплея непосредственно на наглядном изображении. Итеративная визуализация трехмерных моделей позволяет пользователю осматривать объект посредством управления перемещением аппаратов проецирования в моделируемом пространстве	СМ-4, СМ-1420	Саонов, 1985
РЕДГРАФ (СССР)	Автоматизация выпуска чертежно - конструкторской документации	Чертеж представляется в виде сегментированного файла графических элементов. Файл может включать фрагменты — символы хранимых файлов специализированной базы данных. Любой чертеж может быть записан в эту базу, а затем вызван в качестве фрагмента другого чертежа. В системе реализуется иерархическая структура данных. Позволяет идентифицировать элемент чертежа на любом уровне структуры.	Чертеж представляется в виде сегментированного файла графических элементов. Файл может включать фрагменты — символы хранимых файлов специализированной базы данных. Любой чертеж может быть записан в эту базу, а затем вызван в качестве фрагмента другого чертежа. В системе реализуется иерархическая структура данных. Позволяет идентифицировать элемент чертежа на любом уровне структуры.	Электроника 100-25, Фортран, ОС РАФПС	Голиков и др., 1985
РИС (СССР)	Построение простых рисунков (блок-схем) по словесному описанию, приближенному к русской прозе	Благодаря синтаксической избыточности стиль описания рисунка приближается к стилю естественного языка. Для сокращения описаний широко применяется принцип умолчания	ЕС ЭВМ, СИПЛ	Катков и др., 1984	

Наименование системы (страна)	Назначение	Особенности реализации	Аппаратные и программные средства	Источник
CAD dy, CAD dy Corp. (США)	Автоматизация проектирования печатных плат	Возможны: генерация чертежа принципиальной схемы, автоматическое извлечение списка соединений, интерактивное размещение, автоинтерактивная трассировка	IBM PC/XT, AT, ОС MS-DOS	Ойхман, 1986
DRAINAGE Нантская высшая национальная школа механики (ENSM) (Франция)	Автоматизация проектирования дренажной системы на выделенном участке	Начальный этап работы (построение так называемой основы топографического плана) выполняется ЭВМ автоматически. Затем с помощью графического планшета в интерактивном режиме производится размещение секций дренажа и отдельных труб		Гардан и др., 1987
GRAFIXI (фирма GIXI, Франция)	Автоматизация чертежных работ	Визуализация управляется набором параметров, позволяющих определить, каким образом рассматривается предмет и как его следует представить. Возможно задание пользователем меню	Мини-ЭВМ, Фортран, АПЛ	GRAFIXI, 1985

Micro Graph 86 (ФРГ)	Система проектирования печатных плат	Реализована автоинтерактив- ная трассировка. Использует- ется технология «резиновых связей»	IBM PC/XT, ОС CP/M	Ойхман, 1986
PIXAR, (фирма Pixar, США)	Синтез цветных объем- ных реалистичных изо- бражений	Используется задание объек- тов алгоритмом их рандомизи- рованного построения, что по- зволяет сочетать простоту за- дания и небольшой объем ис- ходных данных с реалистич- ностью получаемых изображе- ний	Специализированная ЭВМ	Дьюдни, 1987
SPACE (СССР)	Поддержка трехмерных графических задач в САПР	Для хранения описаний объек- тов используется специальный оперативный архив. Средства архива позволяют идентифици- ровать объект при его форми- ровании программной перемен- ной и в дальнейшем исполь- зовать ее при работе с объек- том	БЭСМ-6, ЕС ЭВМ и СМ ЭВМ, Фортран	Мацокин и др., 1984

построение объекта путем задания операции повышения размерности над ранее определенным объектом, т. е. построение поверхностей и тел вращения и постоянного сечения.

Визуализация осуществляется с помощью синтеза проекций объектов.

Система GRAFIXI. Эта графическая система создана французской фирмой GIXI и предназначена для автоматизации чертежных работ различного профиля. Она реализована на 16-разрядной мини-ЭВМ. Графический терминал Radiance-320, входящий в состав системы, способен генерировать до 256 различных цветов на экране с разрешением 640×512 пикселей. В качестве языков программирования в системе используются АПЛ и Фортран.

Базовое математическое обеспечение системы GRF позволяет поддерживать двух- и трехмерные графические задачи. Заложенные в GRF концепции затрагивают, с одной стороны, описание и создание структур объектов (фигур), с другой — метод их наблюдения и представления. Каждая фигура определяется информацией трех типов:

геометрической, обеспечивающей определение остова объектов; остов может быть выражен линиями, поверхностями и текстами;

информацией связи, определяющей связи между различными точками одного и того же предмета и между различными предметами;

информацией по содержанию, задающей тип линии (цвет, яркость, толщину и т. д.), вид поверхностей (штриховку, цвет, яркость и т. д.), вид текстов (каллиграфию, цвет, толщину и т. д.).

Конкретный вид изображения фигуры определяется набором параметров, задающих способ «рассмотрения» данной фигуры.

Интерактивность системы обеспечивается благодаря использованию интерпретатора АПЛ в качестве интерактивного слоя языка Фортран. Возможность формирования меню, определяемого пользователем, позволяет добиться значительной интеллектуализации системы в конкретной области ее применения. Использование системы GRAFIXI ускоряет процесс построения готового чертежа в 25—100 раз.

Система построения трехмерных графических изображений фирмы PIXAR (США). Система основана на применении специализированного графического компьютера Pixar Image Computer.

В настоящее время система PIXAR — одна из наиболее совершенных графических систем в мире. Для хранения изображения размером 2000×2000 пикселей используется память ЭВМ емкостью 24 Мбайт. Под каждый пиксел выделено 48 бит памяти для хранения информации о различных характеристиках (цвете, интенсивности и т. п.) данного элемента изображения. Быстродействие компьютера — около 40 млн. опер./с. Скорость обмена информацией между блоком отображения графической информации и памятью 480 Мбайт/с позволяет создавать динамические изображения в реальном масштабе времени.

В процессе построения изображений широко практикуется задание объектов «мира» алгоритмом их построения. При этом неявно используются знания законов симметрично-стохастического строения многих природных систем, которые разработчик реализует в алгоритме формирования изображения. Благодаря этому удается синтезировать сложные картины с множеством неповторяющихся фрагментов при относительно небольшом объеме исходных данных. В процессе построения применяются идеи фронтального задания объектов с рандомизированными параметрами. Это позволяет строить изображения, которые хорошо имитируют структуру природных объектов, так как самоподобно-рекурсивные структуры, и в частности фракталы, распространены в окружающем нас мире [Александров и др., 1984а]. При синтезе изображений ряда объектов (в частности, растений) используются концепции теории графов.

Для имитации эффектов освещенности, отражения света некоторыми поверхностями и других используется метод трассировки лучей. При этом прослеживаются траектории большого числа лучей от глаза воображаемого наблюдателя (через серию отражений по законам геометрической оптики) к поглощающей поверхности.

Отечественные системы обработки визуальной информации**9.1. Пакет программ для распознавания рукописной, символьной и графической информации ГРАФИТ**

А. М. Байков, Е. С. Кузин, А. Л. Ламис

9.1.1. Основные возможности. Пакет программ ГРАФИТ предназначен для автоматического распознавания в ЭВМ выполненных от руки нестилизованых изображений, содержащих символьную, графическую и смешанную символьно-графическую информацию. Использовать пакет можно в разнообразных областях, например в таких, как: автоматизация проектирования, АСУ, полиграфия, картография, статистика и учет, программирование и др.

Основными функциональными возможностями пакета ГРАФИТ являются: распознавание нестилизованых рукописных символьных, графических и смешанных символьно-графических изображений;

расширение базового состава распознаваемых объектов;

прослеживание, выделение и аппроксимация линий;

интерактивная проверка и коррекция результатов распознавания;

управление вводом изображений в ЭВМ с устройств ТС 6030 и ЕС 7979.

Пакет ГРАФИТ обеспечивает распознавание и локализацию в поле изображения объектов, описанных в модели проблемной среды. Базовый состав классов распознаваемых объектов включает рукописные заглавные буквы русского и латинского алфавитов, цифры, знаки пунктуации, знаки арифметических и логических операций, спецсимволы. При необходимости эти описания должны добавляться пользователем с ориентацией на решаемую прикладную задачу.

При подготовке входных изображений не требуется придерживаться каких-либо шрифтов, вписывать распознаваемые символы в специальные поля и т. п. Допускается зачеркивание распознаваемых символов (зачеркнутый символ воспринимается как пробел), а также соприкосновение распознаваемых объектов. На изображении могут присутствовать неизвестные системе объекты, в том числе и помехи. Имеется возможность указания подмножества классов, объекты из которых должны распознаваться на входном изображении. Пользователю предоставляется возможность повышать надежность распознавания за счет увеличения времени распознавания и, наоборот, уменьшать время распознавания за счет надежности.

Распознавание может осуществляться как на всем входном изображении, так и в пределах задаваемого прямоугольного фрагмента.

Передача результатов работы пакета для их использования в других программах осуществляется системой файлов на магнитном диске, принятой в ЕС ЭВМ. Имеется возможность вывода результатов работы пакета в наглядном виде на экран алфавитно-цифрового дисплея или графические терминалы.

Пакет программ ГРАФИТ является открытой системой, допускающей расширение состава классов распознаваемых объектов. Расширение осуществляется путем построения признаковых и структурно-метрических описаний добавляемых классов. Признаковые описания строятся автоматически по обучающей выборке. Структурно-метрические описания классов составляются пользователем на специальном языке описания классов. Структурно-метрическое описание класса содержит: 1) перечень структурных элементов, составляющих объекты из данного класса; 2) отношения между структурными элементами, выполнение которых необходимо для отнесения объекта к данному классу.

Структурные элементы могут быть исходными и составными. Исходными являются отрезок, дуга, кольцо, точка. В качестве составных структурных элементов могут использоваться любые объекты из классов, описанных в модели проблемной среды. Составные структурные элементы могут, в свою очередь, быть описаны с помощью как исходных, так и других составных структурных

элементов, что позволяет строить иерархические описания классов распознаваемых объектов.

В качестве отношений используются связи между структурными элементами, которые определяются либо метрическими характеристиками этих элементов (например, «длина больше»), либо их взаимным расположением на изображении (например, «правее», «соприкасается»).

При задании структурных элементов и отношений применяются конкретизирующие параметры, доопределяющие данный структурный элемент или отношение в описании конкретного класса. Для структурных элементов примером конкретизирующих параметров являются параметры, задающие диапазон допустимой ориентации отрезка, для отношений — параметры, задающие предельно допустимое расстояние между указанными характерными точками структурных элементов в отношении «соприкасается».

Конкретизирующие параметры используются также для вычисления значений «качества» данного структурного элемента, выделенного на изображении, и «качества» выполнения данного отношения. Эти оценки необходимы для вычисления общей оценки «качества» распознанного объекта, которая меняется от 0 (при предельном выполнении условий отнесения объекта к данному классу) до 1 (при «идеальном» выполнении этих условий).

Пакет обеспечивает возможность выделения на изображении линий без распознавания образуемых ими графических объектов. Этот режим применяется, например, для выделения контуров на карте, проводников на схеме трассировки печатной платы и т. п.

При выделении линий осуществляется прослеживание до пересечений связанных участков линий входного изображения, длина которых больше задаваемого минимального значения, и разбиение выделенных участков на прямолинейные отрезки. Минимальное значение длины линий и точность аппроксимации указываются пользователем.

В пакете ГРАФИТ предусматривается возможность после работы пакета в режиме распознавания изображений проверить и при необходимости скорректировать результат распознавания объектов, описанных в модели проблемной среды.

В этом режиме на алфавитно-цифровой дисплей выдается прямоугольная область входного изображения, содержащая объект распознавания, координаты этой области, имя распознанного объекта с указанием оценки качества его распознавания или слово «отказ», а для строки символов — часть строки, содержащая проверяемый символ. Местонахождение проверяемого символа в этой строке помечается стрелкой.

По выведенному на алфавитно-цифровой дисплей изображению и контексту из строки пользователь может проверить правильность результата распознавания рассматриваемого объекта и в случае необходимости скорректировать этот результат.

Объекты для проверки предлагаются в следующей последовательности. Сначала выводятся изображения, соответствующие отказам от распознавания, затем — распознанные объекты в порядке увеличения оценки качества распознавания этих объектов. После проверки очередного объекта пользователь может перейти к проверке следующего объекта или выйти из описываемого режима.

Помимо выполнения основных функций пакет ГРАФИТ предоставляет пользователю ряд дополнительных возможностей сервисного характера, связанных с процессом ввода и подготовки входных изображений, созданием и сопровождением банка введенных изображений, выводом на дисплей или АЦПУ как результата распознавания, так и самих распознаваемых изображений, взаимодействием с внешними программами, синтаксическим контролем составляемых описаний классов распознаваемых объектов и др.

Управление режимами работы пакета осуществляется с применением самодокументированного диалога и не требует от пользователя специальной подготовки.

Пакет программ ГРАФИТ может использоваться для распознавания объектов, допускающих построение отображаемых структурно-метрических описаний в терминах исходных структурных элементов (отрезок, дуга, кольцо, точка) и отношений между ними. Изображения распознаваемых объектов должны состоять из контурных линий. Распознавание полутоновых изображений, а также полиграфических и машинописных документов не предусматривается. Допускаются разрывы между структурными элементами распознаваемых объектов. Разрывы на структурных элементах не допускаются.

Строки подлежащего распознаванию текста на исходном документе должны располагаться горизонтально. Для считывающего устройства ТС 6030 высота распознаваемых символов на исходном документе должна быть не менее 4 мм. Устройства считывания изображений должны обеспечивать не менее 10 линий разложения на 1 мм на исходном изображении.

9.1.2. Основы функционирования. Анализ существующих подходов к построению систем автоматического анализа изображений позволяет сделать вывод о том, что общее решение задачи распознавания изображений должно вестись с позиций методов искусственного интеллекта при организации процесса, основанного на таких составляющих, как целостность восприятия, целенаправленность и использование контекста, т. е. с позиций, характеризующих наше сегодняшнее представление о зрительном восприятии человека.

В акте восприятия зрительной информации на всех его этапах взаимодействуют процессы сверху вниз (от понимания к изображению) и снизу вверх (от изображения к пониманию).

Первый шаг восприятия — это формирование установки или гипотезы о содержании изображения. На этом шаге результат (гипотеза) возникает при взаимодействии процесса сверху вниз, разворачивающегося на основе модели среды и текущего результата восприятия, и процесса снизу вверх, основанного на непосредственном, грубом признаковом восприятии.

Второй шаг — это уточнение восприятия или проверка гипотезы. На этом шаге также взаимодействуют оба процесса, т. е. процессы, включающие операции над информацией из модели и операции на изображении. При этом осуществляется целенаправленный поиск, включающий сегментацию изображения и совместную интерпретацию его частей.

Для обеспечения целостности восприятия результаты локальных операций интерпретируются только совместно в контексте целостных фрагментов или в контексте всего изображения.

Итак, главным принципом, положенным в основу разработки пакета ГРАФИТ, является целостность восприятия. В соответствии с этим любой распознаваемый объект рассматривается как «целое», состоящее из частей, связанных между собой определенными отношениями, причем части получают интерпретацию лишь в составе данного «целого».

Помимо совместной интерпретации элементов изображения в составе распознаваемых объектов принцип целостности восприятия предполагает также интерпретацию самих распознаваемых объектов в составе более крупных целостных образований — конструкций смыслового контекста, отображающего те взаимосвязи из внешней задачи, в которых «участвуют» распознаваемые объекты. Примером семантических конструкций, задающих контекст распознавания, могут служить правила синтаксиса языка программирования в задачах автоматического распознавания и ввода в ЭВМ изображений текстов программ. Учет контекста позволяет не только правильно интерпретировать те объекты, изображения которых могут быть отнесены к двум или более классам, но и повысить надежность распознавания всех объектов, задействованных в той или иной семантической конструкции, за счет их целенаправленной и совместной интерпретации.

Таким образом, в соответствии с принципом целостности восприятия в общем случае можно говорить об иерархии уровней интерпретации элементов распознаваемого изображения: от интерпретации в составе самих объектов, под-

лежащих распознаванию, до интерпретации в составе наиболее крупных семантических конструкций, представленных на данном изображении.

В целом процесс распознавания, реализуемый в описываемом подходе, может быть охарактеризован следующим образом. Процесс распознавания изображения является итеративным и предполагает на каждой итерации использование следующих основных операций:

выдвижение гипотезы о присутствии в данном месте изображения представителя того или иного класса распознаваемых объектов;

выдвижение гипотез о том, что данные объекты, присутствующие на изображении, задействованы в той или иной семантической конструкции;

целенаправленная проверка выдвинутых гипотез;

принятие решений (текущих и окончательных) о справедливости той или иной гипотезы.

Выдвижение гипотез осуществляется на основании снимаемых на изображении признаков, характерных для того или иного класса распознаваемых объектов. При этом в качестве признаков в общем случае используются как локальные характеристики, отражающие особенности элементов входного изображения, так и уже распознанные на предыдущих итерациях объекты.

Проверка выдвинутых гипотез сводится в конечном счете к целенаправленному поиску непосредственно на входном изображении совокупности элементов, удовлетворяющих описанию данного класса распознаваемых объектов. Одновременно вычисляется оценка качества подтверждения гипотезы, которая необходима для разрешения неоднозначностей, а также для организации управления процессом распознавания. При этом целостность восприятия достигается следующим образом:

1) сегментация изображения на отдельные элементы, а также интерпретация этих сегментов осуществляются непосредственно в ходе распознавания (этап априорного кодирования отсутствует) и проводится исключительно в рамках целостного образования — проверяемой гипотезы;

2) сегментация изображения и интерпретация получаемых элементов осуществляются целенаправленно и управляются проверяемой гипотезой, причем все результаты интерпретируются совместно;

3) в случае неподтверждения проверяемой гипотезы все полученные в ходе ее проверки результаты, связанные с выделением и интерпретацией тех или иных элементов изображения, аннулируются.

Принятие решения о справедливости той или иной из выдвинутых гипотез осуществляется на основании результатов проверки этих гипотез в соответствии с полученными оценками качества их подтверждения. Принятое решение в общем случае рассматривается как текущее и может быть пересмотрено на последующих итерациях процесса распознавания. Окончательное решение о правильности распознавания (интерпретация) присутствующих на изображении объектов принимается на основании их совместного рассмотрения в составе наиболее крупных конструкций, отражающих семантику внешней задачи.

Организация процесса распознавания по описанной схеме позволяет использовать модель проблемной среды — систему взаимосвязанных знаний, включающую как собственно описания классов распознаваемых объектов, так и смысловой контекст, который обеспечивает правильное и надежное распознавание этих объектов.

При организации процесса распознавания в соответствии с принципом целостности восприятия необходимо учитывать следующие особенности задания классов распознаваемых объектов.

Изображения объектов, относящихся к данному классу, описываются обобщенно образующими (целиком покрывающими) эти изображения частями и пространственными отношениями между ними. Необходимым и достаточным условием адекватности описаний процессу целостного восприятия является возможность построения по ним любых изображений, но только тех, которые могут быть отнесены к данному классу (свойство отображаемости описаний). Как следствие, такие описания будут обладать свойством полноты. Это свой-

ство позволяет в ходе распознавания соотносить с этим описанием изображения любых представителей данного класса, но только такие, которые могут быть отнесены к данному классу.

В описаниях классов не приводится информация о том, чем отличаются представители данного класса от представителей других классов (свойство независимости описаний). Независимость описаний определяет открытый характер модели проблемной среды.

Еще одним свойством модели проблемной среды является иерархичность. Каждый более высокий уровень иерархии содержит описания классов объектов или конструкций контекста в терминах объектов предыдущего уровня и отношений между ними.

Таким образом, модель проблемной среды представляет собой семантическую сеть фрагментов, каждый из которых соответствует некоторому классу распознаваемых объектов (в том числе и конструкциям контекста) и содержит описание объектов данного класса, обладающее свойствами независимости, полноты и отображаемости.

Сложные изображения иерархически описываются через образующие их более простые и далее, вплоть до простейших (исходных) структурных элементов. В описаниях классов перечисляются структурные элементы, образующие (целиком покрывающие) изображения объектов данного класса, и указываются отношения между этими элементами, определяющие принадлежность объекта именно к данному классу. С именами элементов и отношений связывается список параметров, конкретизирующих интерпретацию этих элементов или отношений в каждом конкретном случае. Для каждого исходного элемента имеется процедура выделения этого элемента непосредственно на изображении, для каждого отношения — процедура его проверки.

Помимо структурных описаний классов в модели содержатся признаковые описания, которые используются на этапе выдвижения гипотез и служат для оценки вероятности того, что на изображении, там где были выделены те или иные признаки, присутствует представитель данного класса.

Дальнейшие исследования направлены в первую очередь на повышение надежности, помехоустойчивости и скорости распознавания с учетом главным образом развития средств описания и все более широкого использования контекста. Ведется работа также по созданию средств автоматизированного и в перспективе автоматического формирования структурно-метрических описаний классов распознаваемых объектов.

9.1.3. Реализация. Пакет программ ГРАФИТ может работать на всех моделях ЕС ЭВМ РЯД-3 с операционными системами ОС 6.1 и ОС 7. Для работы пакета требуется оптимизирующий компилятор для языка программирования ПЛ/1. Необходимая емкость оперативной памяти 1 Мбайт.

При 90 классах распознаваемых объектов и среднем значении коэффициента, определяющего соотношение надежность-время, пакет имеет следующие характеристики:

- достоверность правильного распознавания от $4 \cdot 10^{-2}$ до $1 \cdot 10^{-3}$ в зависимости от качества входной информации;

- среднее время распознавания 15 символ/с на ЭВМ ЕС-1066.

В основном варианте применения пакет ГРАФИТ используется совместно с устройствами ТС 6030 и ЕС 7979.

Устройство ТС 6030 считывает изображение в растровом виде с бумажного носителя размером до 594×841 мм. Устройство ЕС 7979 преобразует считанное изображение из растровой в векторную форму и записывает его на магнитную ленту либо передает непосредственно в память ЭВМ ЕС. На вход пакета информация поступает в виде массива прямолинейных отрезков, аппроксимирующих линии входного изображения.

Допускается использование пакета ГРАФИТ и с другими входными устройствами при условии преобразования изображений в векторную форму. В перспективе планируется включение в состав пакета ГРАФИТ программ, обеспечивающих возможность работы пакета с изображениями, считываемыми

и передаваемыми в ЭВМ непосредственно в растровом виде. Разработка завершена в декабре 1986 г., пакет передан в СФАП «Алгоритм».

В 1989 г. заканчивается разработка модифицированной версии пакета. Предполагается внести улучшения, в том числе внешних характеристик (быстродействия, надежности, помехоустойчивости), расширить базовый состав распознаваемых объектов, обеспечить возможность работы на растровом изображении и распознавания изображений объектов, имеющих разрывы на структурных элементах, и др.

9.2. Система технического зрения для анализа сложных трехмерных сцен

В. П. Андреев

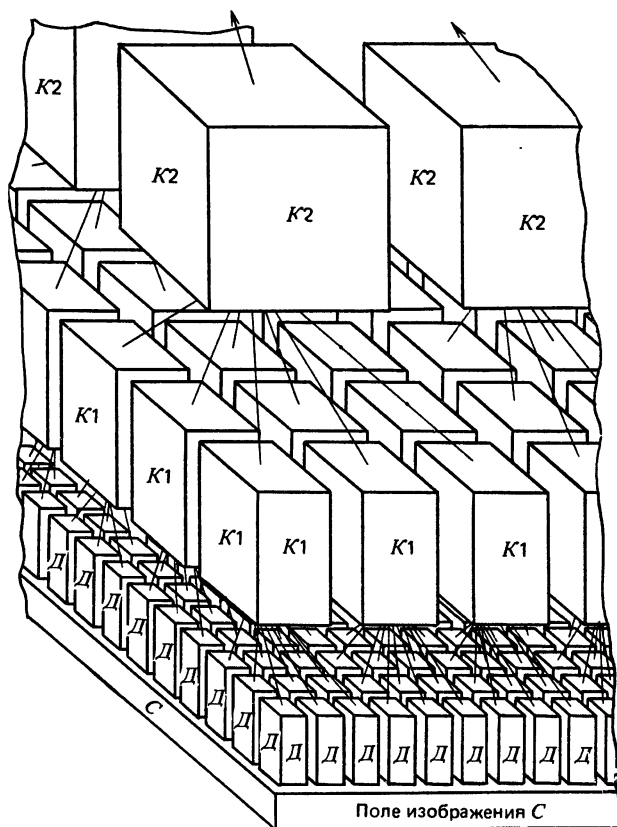
9.2.1. Общие сведения. Система технического зрения (СТЗ) [Андреев, 1981] предназначена для построения графического препарата изображения трехмерной сцены, состоящей из объектов с плоскими поверхностями. На вход СТЗ поступает видеосигнал с промышленной телевизионной установки (ПТУ), на выходе формируется список отрезков прямых линий, аппроксимирующих видимые ребра объектов на изображении — графический препарат. Отрезки задаются координатами своих концевых точек, определенных в плоскости изображения. Графический препарат поступает для анализа в блок распознавания объектов [Андреев и др., 1987].

Система представляет собой комплекс из аппаратных и программных средств. Аппаратные средства состоят из ПТУ, устройства ввода телевизионного изображения в ЭВМ [Андреев и др., 1982] и собственно ЭВМ. Программные средства — это комплекс программ на ЭВМ, реализующих функциональную модель специализированного матричного процессора с иерархической структурой [Андреев, 1979]. В основу модели положена идея параллельной обработки больших объемов видеoinформации, что позволит достичь реального времени для процесса предварительной обработки видеосигнала (сегментации изображения). Задача моделирования сводилась к выявлению работоспособности метода параллельной сегментации изображений реальных трехмерных сцен, полученных с помощью телевизионного датчика. Особенность данного метода сегментации заключается в одновременном использовании в системе как метода выращивания областей, так и метода выделения контуров, что дает возможность увеличить помехоустойчивость системы.

9.2.2. Компоненты и функции системы. Функциональная схема СТЗ представлена на рис. 9.1. На вход системы с ПТУ поступает видеосигнал, который с помощью устройства ввода изображений в ЭВМ (УВИ) подвергается дискретизации и квантованию. Полученное цифровое изображение заносится в буферную память *C*, превращая ее тем самым в некоторое подобие сетчатки глаза человека. Все поле «сетчатки» разбивается на большое число перекрывающихся и одинаковых по форме и размерам участков. Каждый такой участок рассматривается как самостоятельный фрагмент изображения, и отсчеты видеосигнала на этом элементарном фрагменте поступают в специализированный процессор *D*, который вычисляет некоторый набор локальных признаков изображения, например параметры контурной линии, проходящей через фрагмент. При этом выполняется фильтрация шумов. Множество одинаковых процессов *D* образуют первый уровень иерархии.

Второй уровень иерархии также состоит из множества одинаково работающих процессоров-классификаторов *K1*, которые оперируют данными, поступающими от группы смежных процессоров первого уровня. Процессоры третьего уровня иерархии выполняют обработку информации от группы смежных процессоров второго уровня и т. д. Иерархию завершает один процессор *KN*, на выходе которого должны формироваться параметры всех однородных областей, обнаруженных на изображении. Объем обрабатываемой информации от уровня к уровню сокращается, а трудность обработки при этом возрастает, так

Рис. 9. 1. Функциональная схема системы технического зрения



как приходится анализировать все более сложные структуры. За счет распараллеливания процесса обработки на каждом уровне иерархии достигается значительный выигрыш во времени.

Первый уровень иерархии представляет собой набор локальных операторов выделения признаков. Считается, что анализируемые полутонные изображения можно аппроксимировать множеством малых фрагментов, состоящих не более чем из двух областей различной яркости. На дискретном растре каждый такой фрагмент Φ можно представить в виде вектора яркостей $\mathbf{B}_\Phi$ в N -мерном пространстве, где N — число элементов дискретного растра на фрагменте. Для нахождения параметров контура и усредненных значений яркостей областей слева и справа от него используется метод масок [Андреев, 1979]. Маски — это некоторый конечный набор моделей идеального ступенчатого контура, сформированных в виде нормированных векторов $\{\mathbf{H}_k\}$ для $k=1, 2, \dots, K$ в том же базисе, что и $\mathbf{B}_\Phi$. Работа оператора выделения признаков заключается в нахождении

$$G^* = \max_k |G| = \max_k |(\mathbf{B}_\Phi, \mathbf{H}_k)| \text{ для } k = 1, 2, \dots, K.$$

Если эта величина меньше некоторого порога T , то считается, что данный фрагмент содержит одну однородную область, значение яркости которой соот-

ветствует средней яркости по фрагменту B . В противном случае фрагмент содержит две однородные области с границей, параметры которой определяются k -й маской, давшей максимальный по модулю отклик. При этом вычисляются усредненные значения яркостей областей слева и справа от контура B_l и B_r для k -го вектора-модели:

$$\bar{B} = \left(\sum_{i=1}^{m_l} B_{li} + \sum_{i=m_l+1}^N B_{ri} \right) / N,$$

$$\bar{B}_l = \frac{1}{m_l} \sum_{i=1}^{m_l} B_{li} = \bar{B} + \frac{\omega}{A\sqrt{N}} G,$$

$$\bar{B}_r = \frac{1}{m_r} \sum_{i=m_l+1}^N B_{ri} = \bar{B} - \frac{\omega^{-1}}{A\sqrt{N}} G,$$

где B_l и B_r — значения отсчетов видеосигнала («яркостей») в точках дискретного раstra соответственно слева и справа от контура; m_l и m_r — количество элементов k -й модели соответственно слева и справа от контура ($m_l + m_r = N$); $\omega = \sqrt{m_r/m_l}$; A — константа, характеризующая модуль вектора-модели.

Параметры границы, разделяющей однородные области и представляющей собой отрезок прямой линии, описываются коэффициентами нормального уравнения прямой.

Второй уровень иерархии — это набор классификаторов, задача которых состоит в выполнении классификации данных, поступающих от нескольких соседних операторов выделения признаков D . Входные данные — это список отрезков прямых фиксированной длины (сегментов) и список областей, т. е. фрагментов, на которых контуры не обнаружены. Сегменты определяются коэффициентами нормального уравнения прямой, координатами центров соответствующих им фрагментов и значениями средних яркостей слева и справа, а области — усредненными значениями яркостей и координатами центров фрагментов.

Классификатор одновременно анализирует информацию от двух смежных операторов выделения признаков. При этом возможны три случая: оба оператора на своих фрагментах обнаружили однородные области; один оператор обнаружил однородную область, а другой выделил границу между двумя однородными областями; оба оператора выделили границы.

В первом случае классификатор должен определить принадлежность выделенных на фрагментах областей одной однородной области и, если такая принадлежность выявлена, слить эти области в одну, во втором — определить, не сливаются ли какие-нибудь две из трех областей. В третьем случае выполняются сразу две проверки: во-первых, не сливается ли какая-либо пара областей в одну и, во-вторых, принадлежит ли пара сегментов одной границе между двумя однородными областями. Анализ всех этих случаев сводится к решению двух задач: аппроксимации двух областей и аппроксимации двух отрезков прямых.

Обе задачи решаются методом наименьших квадратов. Аппроксимация областей выполняется по формуле

$$B = s_i \bar{B}_l + s_j \bar{B}_j,$$

где $\bar{B}_l$ и $\bar{B}_j$ — средние значения яркостей аппроксимируемых областей; s_i и s_j — коэффициенты, учитывающие площади областей ($s_i + s_j = 1$).

Правило работы классификатора KI при анализе областей: слить две смежные области, если среднеквадратическая ошибка их аппроксимации меньше некоторого порогового значения.

При аппроксимации двух отрезков прямых параметры результирующей прямой определяются следующим образом:

$$\theta_{cp} = u_i \theta_i + u_j \theta_j \text{ для } 0 \leq |\theta_{cp} - \theta_k| \leq 90^\circ, \quad \rho_{cp} = u_i \rho_i + u_j \rho_j,$$

где θ_i и θ_j — углы наклона нормалей аппроксимируемых сегментов; ρ_i и ρ_j — расстояния i -й и j -й прямых соответствующих контурных сегментов от начала координат (коэффициенты нормального уравнения прямой); u_i и u_j — весовые коэффициенты, определяемые длиной соответственно i -го и j -го сегментов ($u_i + u_j = 1$); индекс $k = i, j$.

Правило работы классификатора $K1$ при анализе отрезков линий: слить два отрезка, если среднеквадратическая ошибка их аппроксимации меньше некоторого порогового значения и если обе пары областей по разные стороны от сегментов образуют две однородные области.

Возникает также задача определения координат концевых точек нового аппроксимирующего сегмента. Она решается так. Из четырех концевых точек исходных сегментов опускаются перпендикуляры на аппроксимирующую прямую и вычисляются координаты точек пересечения:

$$x_p = (\rho_{cp} - y_i^* \sin \theta_{cp}) / \cos \theta_{cp},$$

$$y_p = \rho_{cp} \sin \theta_{cp} - (x_i^* \sin \theta_{cp} - y_i^* \cos \theta_{cp}) \cos \theta_{cp},$$

где x_i^* и y_i^* — координаты концевых точек сегментов ($i = 1, 2, 3, 4$).

Точки, расстояние между которыми максимально, будут концевыми точками аппроксимирующего сегмента.

Таким образом, результатом работы каждого классификатора второго уровня иерархии являются списки контурных сегментов, однородных областей и их взаимные пространственные соотношения на участке изображения, контролируемом данным классификатором со своим набором локальных операторов первого уровня иерархии.

Третий уровень иерархии состоит из классификаторов $K2$, которые аналогично классификаторам второго уровня обрабатывают информацию, поступающую от группы смежных классификаторов предыдущего уровня.

Алгоритм работы классификатора $K2$ близок к алгоритму работы классификатора $K1$. Делается простой перебор всех сегментов, полученных на втором уровне иерархии. При этом одновременно рассматриваются два сегмента, взятые из двух смежных классификаторов $K1$. Сегменты считаются принадлежащими одному контуру, если выполняется условие связности: они перекрываются, ошибка их аппроксимации прямой линией меньше заданного порога, ошибка аппроксимации областей по обе стороны от аппроксимирующей прямой также меньше заданного порога. В этом случае сегменты маркируются одинаково. В результате полной обработки всем сегментам, полученным на втором уровне иерархии, будет присвоена метка принадлежности контуру. Выполнив аппроксимацию сегментов с одинаковыми метками и определив координаты концевых точек, получим список отрезков прямых, соответствующих ребрам многогранников. Такой список представляет собой графический препарат исходного изображения.

9.2.3. Знания, используемые в системе. Знания определяют главным образом ограничения, налагаемые на модель внешнего мира. В данном случае «мир» состоит из объектов, ограниченных плоскими поверхностями, т. е. многогранников. Причем таких объектов в сцене может быть несколько. Многогранники на плоском изображении можно представить в виде определенным образом связанных однородных областей, границы которых аппроксимируются отрезками прямыми линиями. Для нахождения этих границ на полутоновом изображении сцены используется масочный оператор выделения контуров. Маска — это модель контура, которая наилучшим образом (в смысле минимума среднеквадратической ошибки) аппроксимирует ступеньку функции яркости, определенную на небольшом фрагменте изображения. Фрагмент содержит 69 элементов ди-

скретного растра исходного изображения, взятых на окне размером 9×9 и попавших в окрестность приблизительно круглой формы.

Система содержит 72 эталонные маски, каждая из которых представлена в виде 69-мерного вектора нормированных коэффициентов, определяющих ориентацию и положение контура на фрагменте. Считается, что контур на элементарном фрагменте может быть аппроксимирован отрезком прямой, ориентированным в одном из 12 направлений, т. е. через каждые 15° , причем в каждом направлении может быть шесть позиций контура на фрагменте (по три, расположенные с равным шагом симметрично от центра фрагмента).

Набор эталонных масок описан в системе как двухмерный массив целых чисел (матрица) размером 72×69 . Дополнительно в систему включены два массива по 72 ячейки с полярными параметрами (θ_i и ρ_i) соответствующих эталонных контурных линий и два массива по 324 ячейки, содержащие координаты центров элементарных фрагментов.

9.2.4. Диалоговые возможности системы. Программная реализация СТЗ на ЭВМ допускает работу в диалоговом режиме, когда возникает необходимость настройки системы на изменяющиеся внешние условия, влияющие на качество получаемого с помощью ПТУ полутонового изображения. Настройка заключается в подборе значений различных порогов, используемых при аппроксимации на всех уровнях иерархии. Настройка выполняется при неизменных внешних условиях (уровень освещенности сцены, расположение источников света, тип фоновой поверхности, класс объектов и т. п.) на типичной сцене. Выбор значений порогов производится субъективно оператором ЭВМ на основе наблюдения результирующего контурного препарата (каждого уровня иерархии), который отображается на экране полутонового дисплея «наложенным» на исходное изображение сцены.

9.2.5. Реализация. На входе СТЗ имеется изображение, представленное в памяти ЭВМ в виде массива размером 128×128 элементов дискретного прямоугольного растра при 64 уровнях квантования видеосигнала. Координаты элементов задаются в декартовой системе в пределах от 1 до 128. В качестве датчика видеосигнала использована промышленная телевизионная установка типа МТУ-1п, которая подключена к ЭВМ через специальное устройство согласования, работающее по принципу стробоскопирования [Андреев и др., 1982].

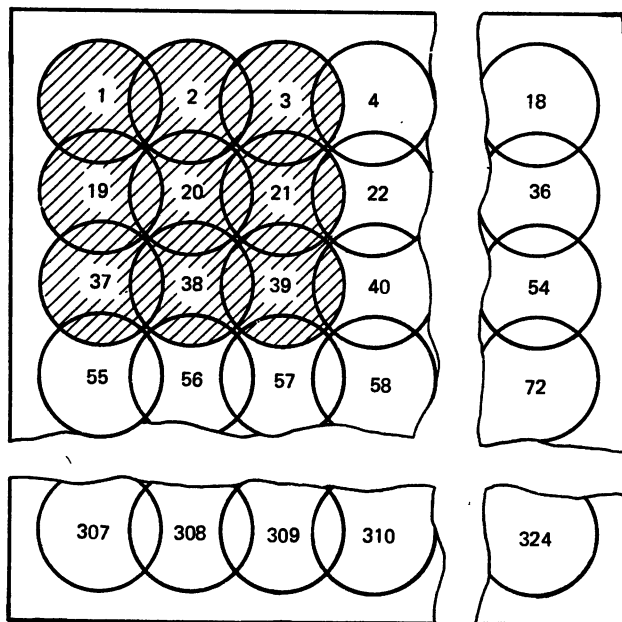
Входное изображение разбивается на 324 элементарных фрагмента (рис. 9.2), каждый из которых содержит 69 элементов дискретного растра (см. п. 9.2.3). Таким образом, все поле значений отсчетов видеосигнала представляется в виде 324 входных 69-мерных векторов яркостей.

Первый уровень иерархии состоит из 324 процессоров, выполняющих вычисление признаков изображения на элементарных фрагментах. При моделировании работы процессора первого уровня иерархии (D) определялось, проходит ли через фрагмент контур. Если контур отсутствовал, вычислялось среднее значение функции яркости на фрагменте; в противном случае находились полярные параметры прямолинейного отрезка, который наилучшим образом аппроксимирует этот участок контура. Кроме того, вычислялись средние значения функции яркости областей слева и справа от аппроксимирующей линии. Эти параметры определялись перемножением входного 69-мерного вектора яркостей на каждый из 72 эталонных 69-мерных векторов. С учетом знака скалярного произведения можно проанализировать 144 ситуации. Параметры изображения вычислялись по тому эталонному вектору, скалярное произведение которого на входной вектор дает максимальное по модулю значение.

Второй уровень иерархии состоит из 36 процессоров (6×6); на вход каждого из них поступают признаки изображения от девяти смежных процессоров первого уровня (см. штриховку на рис. 9.2). Таким образом, все процессоры первого уровня иерархии разбиты на 36 групп.

При моделировании иерархического матричного процессора модель была ограничена тремя уровнями иерархии. Третий уровень модели состоит из одного процессора. На его вход поступают параметры однородных областей, которые были вычислены всеми 36 процессорами второго уровня иерархии. Ввиду огра-

Рис. 9. 2. Пример
входного изобра-
жения



ничений, наложенных на форму тел, на выходе процессора K2 формируются лишь параметры прямолинейных границ однородных областей. В результате строится графический препарат исходного полутонового изображения.

На рис. 9.3 показан процесс его получения. Конечный графический препарат довольно точно соответствует исходному полутоновому изображению. Каждое ребро кубика, за исключением одного, аппроксимировано одним отрезком прямой линии, причем концы этих отрезков приблизительно соответствуют вершинам кубика, линия тени отделена от линии ребра, некоторые ложные сегменты отфильтрованы, образование двух сегментов на одном из ребер кубика вызвано неравномерностью чувствительности фотокатода передающей трубки ПТУ. Такая неравномерность сказывается при низких контрастах контура.

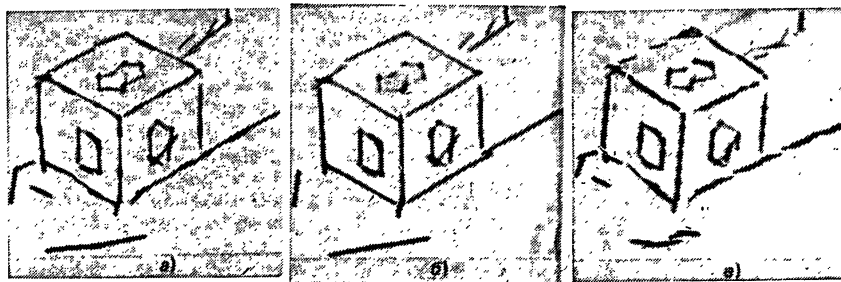


Рис. 9.3. Графические препараты полутонового изображения кубика, полученные в результате работы соответственно первого (а), второго (б) и третьего (в) уровня иерархии

Не всегда удается выделить все контуры. Следовательно, алгоритм идентификации объектов должен работать по неполному графическому препарату. В некоторых случаях можно на втором уровне сегментации восстановить сегмент, ложно принятый за шумовой. Уже на этом уровне удается отделить коллинеарные сегменты, принадлежащие объекту, и тени, что является показателем правильной сегментации. Контуры, которые выделились на гранях кубика, соответствуют изображенным на них цифрам. Вид этого контурного препарата не отражает форму цифр из-за недостаточной разрешающей способности системы. Анализ результатов показывает, что форма контура не искажается, если он проходит не менее чем через два фрагмента.

Таким образом, в результате сегментации двухмерное поле яркостей преобразуется в список границ однородных областей, которые в рассматриваемом случае являются отрезками прямых.

Система технического зрения была использована в экспериментах с машинным зрением, описанных в [Андреев и др., 1987]. Системе были предъявлены сложные сцены, состоящие из многогранников. По изображениям этих сцен, полученным с помощью ПТУ типа МТУ-1п, была выполнена автоматическая идентификация объектов по неполному графическому препарату и определен курс объектов.

9.3. Когнитивная интерактивная графическая система ДСТЧ

А. А. Зенкин

9.3.1. Характеристика когнитивных ИКГ-систем. Когнитивные интерактивные компьютерные графические системы, или когнитивные ИКГ-системы,— это человеко-машинные интеллектуальные системы, ориентированные на создание качественно нового знания в области, прежде всего, фундаментальной науки. Разработка таких систем и методологии их эффективного применения в процессе «расширенного воспроизводства» нового научного знания, а также в качестве уникального инструмента исследования образных интуитивных метапроцедур нашего творческого, правополушарного мышления — новое перспективное научное и практическое направление ИИ. Когнитивные ИКГ-системы образуют в множестве всех интеллектуальных систем своеобразную «экологическую нишу», которая характеризуется следующими основными особенностями.

1. Изначальная ориентация когнитивных ИКГ-систем на целенаправленную активизацию высших творческих метапроцедур образного, интуитивного, правополушарного мышления человека. Подобно тому как с помощью слова мы постоянно и целенаправленно воздействуем на наше вербальное, рационально-логическое, левополушарное мышление, а через него — на эмоции и поведение человека, так и с помощью ИКГ-образов мы можем целенаправленно воздействовать на образное, интуитивное, правополушарное мышление человека для активизации его творческих возможностей и генерации в мозгу человека ответных реакций в форме новых идей и гипотез, т. е. в форме качественно нового знания. Накопление опыта по применению целенаправленного воздействия ИКГ-образов на правополушарное мышление и его анализ впервые открыли возможность объективного научного исследования глубинных механизмов и высших творческих метапроцедур интуитивного, образного мышления. Поэтому особую актуальность приобретают вопросы, связанные с разработкой методологии количественного исследования интуитивных механизмов творческого мышления, основы которой (методологии) впервые сформулированы в работе [Поспелов и др., 1972].

2. Изначальная ориентация когнитивных ИКГ-систем на порождение, на создание качественно нового знания, что составляет одно из главных отличий таких систем от традиционных экспертных систем (т. е. от ЭС, не использующих методы индуктивного вывода) и систем автоматизации научных исследований (САНИ).

3. Интегративный характер когнитивных ИКГ-систем. Процесс познания человеком окружающего мира включает ряд этапов, главными из которых являются: создание нового знания, представление (т. е. фиксация в той или иной форме) и хранение добытого знания, передача готового знания (обучение) и применение знания для решения проблем и в конечном счете для воплощения, овеществления знания в продуктах труда. Опыт, накопленный в процессе прохождения знания по этой цепочке, затем используется для «расширенного воспроизводства» знания, т. е. мы возвращаемся к первому этапу этого бесконечного итерационного процесса, но, естественно, на качественно новом уровне. Традиционные ЭС и САНИ акцентируют свое внимание преимущественно на этапах представления и применения готовых знаний. В рамках же когнитивной ИКГ-системы процесс познания рассматривается и отображается структурно как единый, целостный процесс.

4. Когнитивные ИКГ-системы являются проблемно-ориентированными, т. е. в отличие от естественного языка когнитивный ИКГ-язык, составляющий основу любой когнитивной ИКГ-системы, не является универсальным и его конкретная реализация всегда определяется не только целями его применения, но и спецификой соответствующей проблемной области. Этим же определяется известная специфика изложения материала: описание структуры системы, процедуры построения ИКГ-языка и структуры базы знаний будет дано на конкретных примерах, тесно связанных со спецификой той проблемной области науки, для которой создается данная когнитивная ИКГ-система.

9.3.2. Назначение системы. Система ДСТЧ представляет собой когнитивную человеко-машинную Диалоговую Систему для исследования проблем аддитивной Теории Чисел. Главной особенностью этой системы является использование когнитивной ИКГ, обеспечивающей визуализацию абстрактных математических объектов на экране дисплея в виде особого рода когнитивных ИКГ-изображений и эффективное интерактивное взаимодействие конечного пользователя с ИКГ-изображениями. В процессе взаимодействия когнитивные ИКГ-изображения способны (конечно, лишь при определенных условиях) порождать в голове исследователя такие представления, идеи и гипотезы, которые после их содержательной интерпретации в терминах классической теории чисел и соответствующего обоснования (доказательства их истинности) представляют собой качественно новое содержательно-значимое научное знание в данной проблемной области науки. Другим важным аспектом применения когнитивной ИКГ-системы ДСТЧ является накопление опыта, связанного с пока еще немасштабным и уникальным процессом создания нового научного знания, и анализ этого опыта с целью изучения высших творческих метапроцедур правополушарного интуитивного мышления в условиях реальных процессов решения интеллектуально-трудных научных проблем и технологии порождения качественно нового научного знания.

9.3.3. Описание проблемной области. Классическая теория чисел является, как известно, одним из наиболее трудных и абстрактных разделов современной фундаментальной науки. Первоначально в качестве достаточно узкой, конкретной проблемной области была выбрана так называемая классическая проблема Варинга (КПВ). Эта проблема с точки зрения применения когнитивной ИКГ для генерации качественно нового знания является экстремальной потому, что, во-первых, как и большинство задач теории чисел, она относится к числу наиболее абстрактных, т. е. максимально далеких вообще от всякой графической задачи, и традиционно «интеллектуально трудных» научных проблем; во-вторых, она считается почти исчерпывающе изученной; наконец, в-третьих, эта проблема сыграла важную роль в истории становления современной теории чисел.

Одна из основных проблем аддитивной теории чисел — задача о представлении натуральных чисел $n \geq 1$ суммами s r -х степеней вида

$$n = \sum_{i=1}^s n_i r_i, \quad (1)$$

где все n_i — неотрицательные целые числа, т. е. числа 0, 1, 2 и т. д.

Исследуя эту задачу, английский математик Е. Варинг еще в 1770 г. высказал следующую гипотезу [Хуа Ло-ген, 1964].

Гипотеза Варинга. Для любого фиксированного натурального показателя степени $r \geq 2$ существует наименьшее число слагаемых $g(r)$ — такое, что при любом большем числе слагаемых $s \geq g(r)$ все натуральные числа $n \geq 1$ представимы суммами (1).

Доказательство или опровержение этой гипотезы и составило суть знаменитой КПВ.

9.3.4. Построение когнитивного ИКГ-языка. Итак, в качестве конкретной проблемной области **D** исследования выбрана КПВ. Для построения ИКГ-языка необходимо выбрать множество **A** «аксиом»-высказываний о проблемной области **D** и визуализировать это множество. Поскольку множество **A** должно по определению отражать суть проблемной области **D**, то построению когнитивного ИКГ-языка для **D** должен предшествовать содержательный анализ области **D** с целью выявления ее глубинной сути. Если теперь выражение (1) интерпретировать как проблему связи между аддитивными и мультипликативными свойствами натуральных чисел и если согласиться с тем, что суть какой-либо научной проблемы включает совокупность причин, обуславливающих принципиальные трудности ее решения, то суть КПВ состоит в математических свойствах связи между аддитивными и мультипликативными свойствами натуральных чисел, ибо в математике хорошо известно, что вся трудность (и не только КПВ, но и всей теории чисел) заключается в том, что свойства целых чисел относительно умножения (мультипликативные свойства) с их свойствами относительно сложения (с их аддитивными свойствами) связаны очень сложно [Делоне, 1959].

Именно эфемерность и недоступность для нашего «живого созерцания» этой возведенной в квадрат научной абстракции, а именно абстрактной связи между абстрактными же свойствами аддитивности и мультипликативности натуральных чисел, и является, по-видимому, главной причиной того поистине уникального факта, что такие знаменитые проблемы теории чисел, как проблема простых чисел, проблема Ферма и проблема Гольдбаха, в том, что касается их формулировки, доступны пониманию даже школьников, но полное решение этих проблем до сих пор выходит за рамки возможностей даже современной, вооруженной мощнейшей вычислительной техникой науки.

Когнитивная же ИКГ позволяет эффективно решить задачу визуализации этой «эфемерной» связи. При этом искомое множество **A** «аксиом» — высказываний, определяющих суть проблемной области **D**, оказывается состоящим всего из ... одного-единственного чисто эвристического по своей природе постулата.

Постулат. Подавляющее большинство интересных теоретико-числовых свойств натуральных чисел — этих элементарных базовых объектов проблемной области **D** — определяется, в конечном счете, характером связи между их аддитивными и мультипликативными свойствами.

Опишем алгоритм построения ИКГ-изображений, позволяющих визуализировать эту абстрактную связь.

Обозначим **N** множество натуральных чисел, а $\Delta = \{n', n''\}$ — любой фиксированный отрезок из **N**. Пусть P — произвольный теоретико-числовой предикат, заданный на **N**, так что для любого натурального числа $n \in \mathbf{N}$ имеет место либо $P(n)$, либо не- $P(n)$. Отрезок Δ будем изображать на экране графического дисплея прямоугольной таблицей $K \times L$ квадратиков δ_{ij} , где i — номер строки; j — номер столбца таблицы, в которых находится данный квадратик. Каждому квадратику δ_{ij} соответствует изображаемое им натуральное число $n_{ij} = n' + (i-1) \times L + j$. Количество столбцов L данной таблицы назовем модулем ИКГ-изображения. Условимся квадратик δ_{ij} закрашивать в зеленый цвет, если $P(n_{ij})$, и в красный, если не- $P(n_{ij})$. На рисунках, которые приводятся ниже, это белый и черный цвета соответственно.

На основе этого алгоритма система ДСТЧ строит на экране дисплея ИКГ-образ отрезка Δ в виде прямоугольника $K \times L$ с характерной, зависящей от P , L и теоретико-числовых свойств $n \in N$ мозаикой зеленых и красных квадрати-ков δ_{ij} .

Сопоставляя каждому выводимому на экране квадратику δ_{ij} некую звуковую метку, зависящую от его цвета и положения, можно получить музыкальную фонограмму соответствующего математического объекта (подмножества из N), по своему звучанию нередко напоминающую классическую музыку. Если же, напротив, структуру какого-либо математического объекта (например, теоремы Эйлера или теоремы Гаусса) использовать в качестве фильтра, налагаемого на классическую мелодию, нередко удастся услышать некоторые современные популярные мелодии. Однако и в том и в другом случае ритмика такой фонограммы определяется (задается) именно структурой изображаемого математического объекта, так что некоторые характерные инварианты этой (абстрактной) математической структуры можно улавливать и идентифицировать не только визуально, но и на слух.

Следуя традиции, например по аналогии с общеизвестными РЕНТГЕНОграммами, и по причинам, исторически достаточно очевидным, назовем такие цветомузыкальные ИКГ-изображения абстрактных математических объектов ПИФАГОРОграммами, или короче пифограммами таких объектов.

Математический смысл таких пифограмм состоит в следующем. В суммах (1) фиксируем произвольные значения $r \geq 2$, $s \geq 1$ и вводим предикат P согласно условию: если натуральное число $n \in \Delta$ представимо суммой (1), то имеет место $P(n)$, в противном случае — не $P(n)$. В соответствии с только что приведенным алгоритмом построим пифограмму отрезка $\Delta = [n', n'']$. Поскольку значение $P(n)$ зависит от представимости n , то цвет квадратиков на этой пифограмме моделирует, отображает аддитивные свойства натуральных чисел. С другой стороны, в любой j -й столбец попадают все числа $n - (n' - 1) \equiv j \pmod{L}$, так что положение любого квадрата на нашей пифограмме моделирует или отображает мультипликативные свойства натуральных чисел. Следовательно, характерная мозаика пифограммы в целом визуализирует именно абстрактную связь между абстрактными теоретико-числовыми свойствами аддитивности и мультипликативности натуральных чисел. Меняя модуль L ИКГ-изображения, получаем уникальную возможность исследовать поведение этой абстрактной связи в динамике.

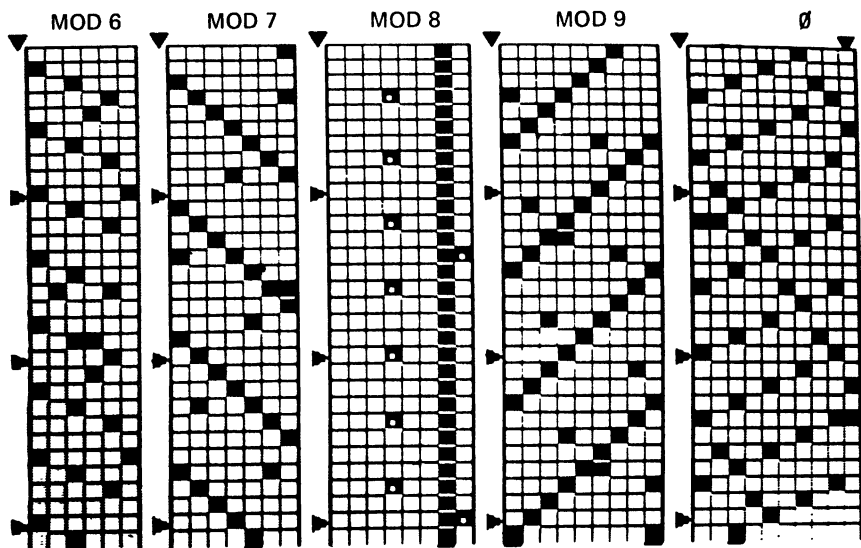
Как уже подчеркивалось, именно характер этой связи между аддитивными и мультипликативными свойствами натуральных чисел определяет специфические трудности и составляет глубинное содержание многих известных и до сих пор нерешенных проблем теории чисел. Но если визуализировать причину достаточно обширного класса математических явлений, то вместе с этой причиной будут одновременно визуализированы и некоторые ее следствия, среди которых могут оказаться и те, которые представляют собой новые математические факты и утверждения. Рассмотрим конкретный пример одного из допустимых преобразований и теоретико-числовой интерпретации визуальных особенностей пифограмм.

В 1801 г. Гаусс доказал свою знаменитую теорему о представимости натуральных чисел суммами трех квадратов неотрицательных целых чисел (см., например, [Хуа Ло-ген, 1964]).

Теорема Гаусса. Суммами (1) трех квадратов неотрицательных целых чисел представимы все натуральные числа, кроме чисел вида

$$n = 4^l(8k+7), \quad k, l = 0, 1, 2, \dots \quad (2)$$

В суммах (1) фиксируем $r=2$, $s=3$ и для отрезка $\Delta = [1, n'']$ с помощью приведенного выше алгоритма строим серию пифограмм (мультифильм) с переменным модулем $L=6, 7, 8, 9$ (рис. 9.4). Как нетрудно видеть, среди этих пифограмм выделяется своей простотой, симметричностью и, если так можно сказать, математической эстетичностью пифограмма по mod 8. Проведем ее расшифровку (интерпретацию) в терминах теории чисел. Основным элементом, глав-



K. F. Gauss'es Theorem (1801)

Рис. 9.4. Фрагмент мультфильма про классическую теорему Гаусса о суммах трех квадратов неотрицательных целых чисел

ная визуальная особенность этого ИКГ-изображения — красный (на рис. 9.4 черный) столбец № 7. В силу самого алгоритма построения пифограмм это позволяет сформулировать следующий визуально-очевидный факт (машинную гипотезу).

ИКГ-утверждение 1. Все квадратики столбца № 7 — красные, т. е. все натуральные числа вида

$$n = 8k + 7, \quad k = 0, 1, 2, \dots, \quad (3)$$

не представимы суммами трех квадратов неотрицательных целых чисел.

В когнитивной ИКГ-системе ДСТЧ имеется подсистема ВЫМЕТАНИЕ, которая автоматически определяет среди элементов любого заданного множества любые подпоследовательности типа арифметических и геометрических прогрессий, помечает по желанию пользователя соответствующие квадратики пифограмм, являющиеся элементами таких подпоследовательностей, визуальными (на рис. 9.4 это белые метки внутри черных квадратиков) и звуковыми метками и выдает параметры соответствующих подпоследовательностей. Как видно на пифограмме по mod 8, в результате работы подсистемы ВЫМЕТАНИЕ все красные (черные) квадратики, не являющиеся элементами столбца № 7, оказываются помеченными как элементы геометрической прогрессии $a_0 4^l$, $l = 1, 2, \dots$, где a_0 — одно из чисел вида (3). Это позволяет дополнить ИКГ-утверждение 1 следующим визуально-очевидным фактом (машинной гипотезой).

ИКГ-утверждение 2. Все красные квадратики, не стоящие в столбце № 7, соответствуют натуральным числам вида

$$n = (8k + 7) \cdot 4^l, \quad l = 1, 2, \dots \quad (4)$$

Объединяя теперь эти два ИКГ-утверждения, т. е. выражения (3) и (4), получаем выражение (2), так что проведенная теоретико-числовая интерпретация двух визуально-очевидных фактов на пифограмме отрезка $\Delta = [1, n']$ по mod 8 оказывается тождественной классической теореме Гаусса.

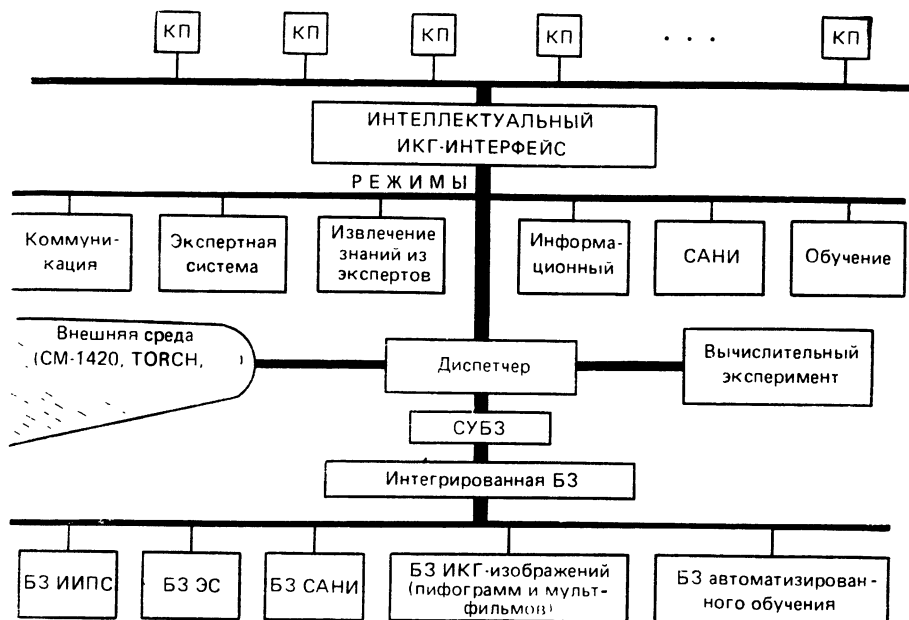


Рис. 9.5. Структурная схема человеко-машинной когнитивной ИКГ-системы ДСТЧ

Итак, построен когнитивный ИКГ-язык. Элементарными символами этого двумерного графического языка являются помеченные цветом (красные и зеленые) и звуком (например, «до» и «соль») квадратик δ_{ij} . Из этих элементарных символов по указанному выше алгоритму строятся элементарные графические термины (слова или выражения I уровня) — пифограммы.

На множестве таких термов-пифограмм заданы допустимые графические операции: масштабирование (изменение размеров квадратиков δ_{ij}), позиционирование (размещение пифограммы на экране дисплея), изменение цветовых и музыкальных меток по различным условиям (выделение фрагментов — особенностей пифограмм), а также операции изменения модуля изображения и параметров r и s в суммах (1). С помощью этих операций можно строить уже более сложные выражения II уровня («фразы» формулы) — конечные последовательности пифограмм (как в пространстве, так и во времени), т. е. цветово-музыкальные мультфильмы. Как показано на приведенном только что примере, такой когнитивный ИКГ-язык имеет непротиворечивую содержательную интерпретацию в области аддитивной теории чисел. При необходимости можно дать более формализованное описание этого графического когнитивного ИКГ-языка.

9.3.5. Общая структура когнитивной ИКГ-системы ДСТЧ, ее основные компоненты и реализуемые ими функции. На рис. 9.5 приведена структурная схема когнитивной ИКГ-системы ДСТЧ. Перечислим кратко основные компоненты этой системы и реализуемые ими функции:

КП — конечный пользователь (в частности, эксперт) за терминалом сетевой персональной ЭВМ;

ИНТЕЛЛЕКТУАЛЬНЫЙ ИКГ-ИНТЕРФЕЙС — совокупность подсистем, обеспечивающих максимально «дружелюбное» общение коллектива конечных пользователей с исследуемой проблемой (или проблемами) на базе когнитивного ИКГ-языка, применяемого на всех этапах такого общения.

Основные режимы:

КОММУНИКАЦИЯ обеспечивает обмен информацией, сообщениями, задачами и т. п. между различными КП, а также пересылку задач на вычислительные мощности внешних ЭВМ (например, головной ЭВМ TORCH, CM-1600 и др.) и получение результатов;

ЭС обеспечивает проведение исследований визуальных особенностей пифограмм с помощью системы продукционных правил, хранящихся в БЗ ЭС, формулировку машинных теоретико-числовых гипотез и проверку их содержательной значимости и новизны;

ИЗВЛЕЧЕНИЕ ЗНАНИЙ ИЗ ЭКСПЕРТОВ обеспечивает проведение анализа коллективом экспертов визуальных особенностей пифограмм, выявление взаимосвязей этих особенностей с известными экспертам фактами и утверждениями теории чисел, формулировку взаимосвязей в виде продукционных правил, проверку, обоснование и запись этих правил в БЗ ЭС;

ИНФОРМАЦИОННЫЙ обеспечивает автоматический (например, по ключевым словам) или визуальный поиск и выдачу любой информации, хранящейся в БЗ Интеллектуальной ИПС (ИИПС) по данной проблемной области;

САНИ обеспечивает построение теоретико-числовых объектов и их пифограмм, а также интерактивное исследование их математических свойств;

ОБУЧЕНИЕ обеспечивает обучение КП по данному разделу аддитивной теории чисел как в рамках классических, известных результатов, так и на базе конкретных примеров порождения нового знания в форме «мультфильмов с субтитрами»;

ДИСПЕТЧЕР обеспечивает управление взаимодействием всех блоков, подсистем и режимов работы системы ДСТЧ;

СУБЗ осуществляет управление взаимодействием КП с различными БЗ, а также взаимодействием между БЗ, их пополнением и модификацией;

ИНТЕГРИРОВАННАЯ БЗ — система взаимосвязанных БЗ, из которых основными являются следующие:

БЗ ИИПС содержит в основном декларативную информацию о рассматриваемой проблемной области аддитивной теории чисел: постановка задач, описание основных подходов и методов их решения, основные результаты по КПВ и ее классическим обобщениям, краткие рефераты и т. п., постоянно пополняется новыми результатами;

БЗ ЭС содержит семантическую сеть основных результатов и обобщений КПВ и постоянно пополняемую систему продукционных правил типа

ЕСЛИ <на данной пифограмме имеется такая-то визуальная особенность>

ТО <по-видимому, имеет место такая-то содержательно-значимая машинная теоретико-числовая гипотеза>

БЗ САНИ — это иерархическая система математических моделей всевозможных подзадач и обобщений КПВ;

БЗ ИКГ-ИЗОБРАЖЕНИЙ (ПИФОГРАММ) — файловая система отдельных пифограмм и целых мультфильмов в двоичной кодировке на ГМД по классическим и неклассическим проблемам аддитивной теории чисел;

БЗ АВТОМАТИЗИРОВАННОГО ОБУЧЕНИЯ — совокупность обучающих модулей по различным проблемам аддитивной теории чисел, содержащих соответствующие мультфильмы о динамике математических структур с необходимыми текстовыми пояснениями, с примерами порождения новых продукционных правил и новых результатов на основе ИКГ-подхода.

Существует тесная взаимосвязь различных режимов работы в рамках когнитивной ИКГ-системы ДСТЧ. Так, в режиме ИЗВЛЕЧЕНИЕ ЗНАНИЙ ИЗ ЭКСПЕРТОВ можно на одном из его этапов воспользоваться ИНФОРМАЦИОННЫМ режимом, для того чтобы узнать, какого типа продукционные правила БЗ ЭС используются для решения данного типа задач, а на другом этапе можно воспользоваться режимом САНИ для построения новых математических объектов и их пифограмм. То же относится и к взаимосвязям между различными БЗ. Например, если рассматривается новая задача, для которой созда-

ется новая математическая модель, то эта модель заносится в БЗ САНИ, а соответствующая ей пифограмма или мультфильм — в БЗ ИКГ-ИЗОБРАЖЕНИЙ. Для анализа этой новой пифограммы используются производционные правила из БЗ ЭС; если же ни одна из существующих продукций не является эффективной, т. е. не позволяет получить новое знание, то в режиме ИЗВЛЕЧЕНИЕ ЗНАНИЙ проводится коллективная экспертиза исследуемой пифограммы. Если в процессе такой экспертизы создается новая продукция, порождающая новое знание, то такая продукция заносится не только в БЗ ЭС, но вместе с соответствующей математической моделью, пифограммой (или мультфильмом) и пояснительным текстом в виде стандартизированного по определенным правилам обучающего модуля — в БЗ АВТОМАТИЗИРОВАННОГО ОБУЧЕНИЯ. В свою очередь, последней в любое время доступны знания, хранящиеся в других БЗ. Такого рода относительная автономия различных БЗ при наличии оперативной взаимосвязи между ними позволяет существенно сократить общие затраты машинного времени на решение проблем и повысить эффективность работы КП.

9.3.6. Типы знаний и способы их представления в системе ДСТЧ. Как уже отмечалось, когнитивные ИКГ-системы являются интегрированными интеллектуальными системами, охватывающими все основные этапы процесса познания: представление и хранение добытого знания, передачу (обучение) и применение (овеществление) готового знания, систематизацию и структуризацию опыта и создание на его основе нового знания. Главной знаниепорождающей фигурой в рамках когнитивной ИКГ-системы является человек, в голове которого и создается новое значение в процессе его (человека) общения уже не столько с ЭВМ, сколько с исследуемой проблемой общения, основанного на языке ИКГ-изображений сути, глубинного содержания этой проблемы. В когнитивной ИКГ-системе ДСТЧ можно выделить следующие типы используемых знаний и соответствующие им формы представления знаний:

декларативные знания о проблемной области, основная часть которых представляется в традиционной буквенно-цифровой форме и хранится в виде файлов математических текстов в БЗ ИИПС;

экспертные знания представляются в основной системой производционных правил, позволяющих интерпретировать визуальные (и музыкальные) особенности ИКГ-изображений на профессиональном теоретико-числовом языке;

процедурные знания о теоретико-числовых объектах, о методах конструирования и исследования их математических свойств на базе вычислительного эксперимента; языком представления таких знаний является профессиональный язык математических моделей, которые в форме системы соответствующих взаимосвязанных программных модулей (процедур) хранятся в БЗ САНИ;

цветомузыкальные ИКГ-знания о структуре и свойствах абстрактных математических объектов, представленные в форме пифограмм; упорядоченные последовательности пифограмм (мультфильмы) хранятся в БЗ ИКГ-ИЗОБРАЖЕНИЙ;

обучающие знания — это определенным образом структурированные обучающие модули, которые включают в себя знания всех вышеописанных типов; эти модули имеют вид коротких сценариев самого процесса порождения нового знания и в форме отдельных файлов хранятся в БЗ АВТОМАТИЗИРОВАННОГО ОБУЧЕНИЯ.

Опишем динамику взаимосвязей между различными языками представления знаний в когнитивной ИКГ-системе ДСТЧ на конкретном примере процесса порождения нового теоретико-числового знания.

Пример. Визуальный конструктивный критерий строгого доказательства одного класса общих математических утверждений. В системе ДСТЧ на одном из этапов исследования КПБ было получено следующее декларативное знание (фрагмент текстового файла из БЗ ИИПС): «Ранее (дата) был открыт («виден») принципиально новый бесконечный класс так называемых инвариантных

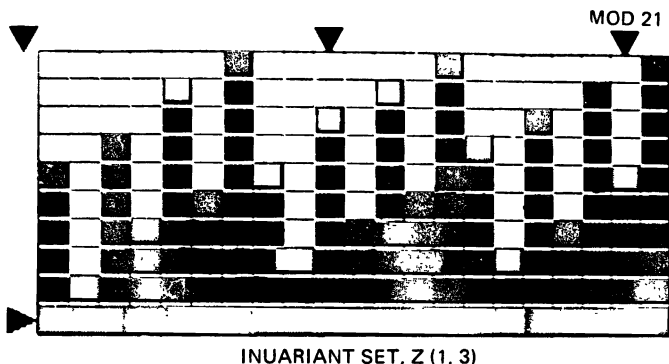


Рис. 9.6. Пифограмма инвариантного множества $Z(1, 3)$ по одному из оптимальных модулей (mod 21)

множеств $Z(r)$ натуральных чисел и была установлена их математическая природа:

$$Z(r) = \left\{ n \geq 1 : n \neq \sum_{i=1}^s (n_i r^i - 1) \text{ при любых } s \geq 1, n_i \geq 1 \right\}.$$

Приведенное в этом декларативном знании определение множеств $Z(r)$ порождает соответствующее процедурное знание — программный модуль в БЗ САНИ, который реализует вытекающий из этого определения алгоритм перечисления всех элементов множества $Z(r)$ для любого фиксированного $r \geq 2$. Однако очевидно, что данный алгоритм не решает вопроса о том, содержит данное множество $Z(r)$ последний элемент или не содержит, т. е. конечно оно или бесконечно, и если оно конечно, то сколько элементов оно содержит. Поэтому следующий этап — порождение цветомузыкального ИКГ-знания.

На рис. 9.6 приведена пифограмма множества $Z(3)$ на отрезке $\Delta = [1, 200]$. Здесь предикат P означает «быть элементом множества $Z(3)$ ». При анализе этой пифограммы сразу возникает вопрос: а все ли элементы множества $Z(3)$ представлены на пифограмме?

Для ответа на этот вопрос рассмотрим следующую визуальную особенность данной пифограммы. Последовательно перенумеруем все зеленые (на рис. 9.6 белые) и только зеленые квадратики пифограммы, т. е. только элементы множества $Z(3)$, слева направо и сверху вниз и каждому такому зеленому квадратику с номером $n = 1, 2, \dots, 75$ сопоставим число $l(n)$ — количество следующих за ним подряд красных квадратиков. В таком случае, например, для $n = 1, 2, 5, 8, 9, 12, 15, 16, 19, \dots$ все $l(n) = 0$, а для $n = 6, 13, 20$ все $l(n) = 1$. С ростом n число $l(n)$ тоже возрастает, хотя и не монотонно. Наконец, для $n = 72, 73$ и 74 все $l(n) = 6$, а для $n = 75$ имеем $l(75) > 6$. Возникает подозрение, что число 6 пороговое в том смысле, что 75-й зеленый квадратик последний (наибольший) элемент множества $Z(3)$. Раскрывая в данном конкретном случае (для $r = 3$) математический смысл этого конкретного порогового числа с помощью следующей цепочки эвристик:

$$6 \rightarrow 8 \rightarrow 2^3 \rightarrow 2^r \rightarrow 2^r - 1,$$

получаем экспертное знание в виде общего утверждения (продукции):

ЕСЛИ (для любого $r \geq 2$ на пифограмме соответствующего множества $Z(r)$ найдется элемент из $Z(r)$ с номером n^* , таким, что для этого элемента $l(n^*) \geq 2^r - 1$),

ТО (элемент с номером n^* является последним (наибольшим) элементом множества $Z(r)$, т. е. соответствующее инвариантное множество $Z(r)$ конечно и представлено на данной пифограмме всеми своими элементами).

Эта продукция устанавливает прямую связь между непосредственно воспринимаемой визуальной (и, добавим, музыкальной) особенностью ИКГ-изображения — увеличением длины «красных» отрезков и отнюдь не тривиальным математическим утверждением о конечности соответствующего декларативно определенного множества $Z(r)$, т. е. общим утверждением о том, что все остальные, не попавшие в кадр натуральные числа не принадлежат множеству $Z(r)$. Она и является принципиально новым научным знанием в области аддитивной теории чисел, хотя пока что и на уровне лишь весьма правдоподобной машинной гипотезы. Однако после ее доказательства [Зенкин, 1983] это продукционное правило превращается в эффективное и конструктивное средство визуального, но абсолютно строгого доказательства конечности множеств $Z(r)$ для любых r . Построенная таким образом и доказанная продукция становится элементом БЗ ЭС, а весь только что описанный процесс порождения нового знания (соответствующим образом оформленный, с замечаниями, пояснениями, отступлениями и дополнениями, а также снабженный математикой, логикой, пифограммами и мультфильмами) превращается в обучающий модуль БЗ АВТОМАТИЗИРОВАННОГО ОБУЧЕНИЯ.

Примечание. Основная трудность, возникающая при разработке экспертных систем, заключается, как известно, в извлечении из экспертов тех знаний, которые отражают их индивидуальный профессиональный опыт и обычно не публикуются. Такие знания являются достоянием специалистов высшей квалификации и, как правило, неохотно сообщаются инженерам по знаниям. Видимо, нужно подумать об условиях, при которых такой специалист был бы заинтересован в передаче своего опыта [Поспелов, 1988]. Проводимый в рамках когнитивной ИКГ-системы именно с целью генерации качественно нового знания (в частности, качественно новых продукций) коллективный экспертный анализ ИКГ-изображений, на которых в максимально наглядной форме отображается суть исследуемой проблемы, позволяет снять некоторые традиционные психологические барьеры, порождающие незаинтересованность эксперта в передаче своего опыта. В частности, сама процедура такого коллективного анализа визуализированной сути исследуемой проблемы создает обстановку активного, очного (что очень важно!), творческого соревнования. Новое знание, которое при этом рождается в голове того или иного конкретного эксперта, просто не успевает обрести психологической атрибутикой «личной собственности», составляющей одно из главных препятствий в передаче экспертами своего опыта инженерам по знаниям.

Емкость БЗ ИИПС составляет в настоящее время около 20 Кбайт, оптимальная ее емкость будет, по-видимому, составлять 2—3 Мбайт. Постоянно пополняемая в процессе эксплуатации системы ДСТЧ БЗ ЭС содержит около 30 продукций и занимает 12 Кбайт. БЗ САНИ содержит около 20 программных модулей для самых различных задач аддитивной теории чисел и занимает около 15 Кбайт. БЗ ИКГ-ИЗОБРАЖЕНИЙ — наиболее емкая база знаний и занимает уже свыше 10 Мбайт. Наконец, БЗ АВТОМАТИЗИРОВАННОГО ОБУЧЕНИЯ содержит пока всего три обучающих модуля и занимает около 600 Кбайт. Оптимальная методология создания таких обучающих модулей находится в стадии разработки.

9.3.7. Языковые и диалоговые возможности системы ДСТЧ. Основными языками, используемыми в системе ДСТЧ при различных режимах работы, являются традиционный текстово-формульный профессиональный математический язык, содержательный язык продукционного типа, язык математических модулей (процедур) и когнитивный ИКГ-язык.

Типы входных и выходных высказываний и ограничения на них.

Текстово-формульный язык. Входными высказываниями являются, как правило, естественно-языковые выражения и целые числа-указатели рубрик иерар-

хических меню; выходными высказываниями — строки профессионального математического текста. Поскольку, с одной стороны, в любой ПЭВМ сегодня имеются средства генерации произвольных символов, а с другой, — выходные высказывания адресуются человеку, никаких принципиальных ограничений на характер или структуру таких высказываний не налагается.

Содержательный язык продукции. Антецедент (условие) продукции — это, как правило, естественно-языковое и (или) профессионально-математическое описание визуальных особенностей какого-либо термина (пифаграммы) или выражения (мультифильма) когнитивного ИКГ-языка консеквент (следствие) продукции — высказывание обычно на профессиональном математическом языке; в настоящее время продукция адресована человеку и применяется в основном на содержательном уровне, поэтому принципиальных ограничений на высказывания данного типа не возникает.

Язык математических моделей. Входными высказываниями являются последовательности имен и целых чисел, включающих названия проблемы, задачи, подзадачи, процедуры, а также значения параметров соответствующих программных модулей; выходными «высказываниями» являются одномерные массивы нулей и единиц, кодирующих свойства конструируемых математических объектов. Существуют очевидные ограничения на длину выходных «высказываний».

Когнитивный ИКГ-язык. Входными высказываниями являются имя файла и совокупность настроечных параметров для пифаграммы теоретико-числового объекта; выходное «высказывание» — цветозвуковое ИКГ-изображение (отдельная пифаграмма или мультифильм). На выходные высказывания такого типа налагаются ограничения, связанные с разрешающей способностью графического экрана.

Структура диалога. В рамках когнитивной ИКГ-системы ДСТЧ диалог реализуется в двух формах.

Многоуровневое меню. Указание раздела в меню данного уровня инициирует или вызов меню следующего уровня, или возврат в меню предыдущего уровня, или реализацию требуемого режима работы системы.

Распределенный диалог. Вывод на экран текстовых или графических визуальных сообщений для пользователя или выдача звуковой информации о состоянии и процессе прохождения задачи (например, звуковое помечивание различного тона системы вложенных циклов, звуковая индикация отдельных этапов решения задачи), а также запросы на ввод информации пользователем распределены по всему «телу» программного комплекса ДСТЧ. Структура сообщений и запросов постоянно оптимизируется разработчиком или частично автоматически в процессе решения конкретной задачи с учетом психологических особенностей и уровня профессиональной подготовки конкретного КП.

9.3.8. Возможности настройки системы на другие предметные области. Когнитивные ИКГ-системы являются существенно проблемно-ориентированными. Это объясняется особенностями когнитивных ИКГ-языков, составляющих методологическую основу таких систем, поскольку графические языки, по определению, должны выражать суть, а значит, и отражать специфику именно той проблемной области, для которой они конструируются. Но это не означает, что когнитивные ИКГ-языки должны быть узкоспециализированными. Об этом свидетельствует опыт применения системы ДСТЧ: первоначально она создавалась для исследования локальной предметной области — КПВ. Однако поскольку положенный в основу реализации когнитивного ИКГ-языка системы ДСТЧ Постулат о связи между аддитивными и мультипликативными свойствами натуральных чисел имеет общий характер, то область применимости данного ИКГ-языка (а вместе с ним и самой системы ДСТЧ) оказалось возможным расширить практически на всю аддитивную теорию чисел и с позиций единого подхода исследовать такие существенно различные задачи аддитивной теории чисел, как КПВ и ее всевозможные классические и неклассические обобщения, проблему простых чисел, классические проблемы Ферма, Гольдбаха, Гольдбаха — Варинга и т. д. Существует предположение о том, что систему

ДСТЧ можно без принципиальных модернизаций настроить на любую предметную область математики, аксиоматика которой изоморфна теории целых чисел. Это и определяет достаточно широкую область приложений когнитивной ИКГ-системы ДСТЧ и лежащего в ее основе ИКГ-подхода к исследованию абстрактных математических объектов средствами когнитивной ИКГ.

9.3.9. Реализация когнитивной ИКГ-системы ДСТЧ. Система ДСТЧ реализована на базе сети персональных ЭВМ TORCH (ОС СРМ); один из ее вариантов реализован также на ЭВМ СМ-1420 и СМ-1600 (ОС РАФОС). Основные языки программирования — Бейсик, Фортран. Требуемая емкость ОЗУ 64 Кбайт (без интегрированной БЗ). Средние временные характеристики: построение одного ИКГ-изображения отрезка натуральных чисел длиной 3000 — около 2 мин; построение математического объекта (при длине отрезка в 5000 чисел) — около 2 ч при быстройдействии 100 000 опер./с.

В когнитивной ИКГ-системе ДСТЧ получен ряд принципиально новых научных результатов в области аддитивной теории чисел: построено нетривиальное обобщение КПВ, открыт новый бесконечный класс так называемых инвариантных, двухпараметрических множеств натуральных чисел обобщенной проблемы Варинга и исследованы их основные математические свойства, аналогичные множества открыты в случае обобщения КПВ, предложенного В. И. Нечаевым [Нечаев, 1953], и в классической проблеме Гольдбаха-Варинга, доказана конечность всех таких инвариантных множеств, обобщены классические теоремы Лагранжа и Вифериха и даны их доказательства, доказано обобщение классической теоремы Гильберта—Варинга и др. [Зенкин, 1983; 1987].

В заключение следует подчеркнуть, что не только возможность такого рода обобщений многих классических результатов и их математическая формулировка, но и сами идеи строгих доказательств основных теорем были подсказаны системой ДСТЧ и увидены впервые именно при анализе цветомузыкальных мультфильмов о свойствах соответствующих абстрактных теоретико-числовых объемов.

Глава 10.

Теоретические аспекты машинного перевода

А. Д. Бакулов, Н. Н. Леонтьева

10.1. Назначение машинного перевода

Машинный перевод (МП), или *автоматический перевод* (АП), — интенсивно развивающаяся область научных исследований, экспериментальных разработок и уже функционирующих систем (СМП), в которых к процессу перевода с одного естественного языка (ЕЯ) на другой привлекается ЭВМ. СМП открывают быстрый и систематический доступ к информации на иностранном языке, обеспечивают оперативность и единообразие в переводе больших потоков текстов, в основном научно-технических. Работаящие в промышленном масштабе СМП опираются на большие терминологические банки данных и, как правило, требуют привлечения человека в качестве пред-, интер- или постредактора. Современные СМП, в особенности те, которые опираются при переводе на базы знаний в определенной предметной области, относят к классу систем искусственного интеллекта (ИИ).

Основные сферы использования МП:

1. В отраслевых службах информации при наличии большого массива или постоянного потока иноязычных источников. Если СМП используются для выдачи сигнальной информации, постредактирование не требуется.

2. В крупных международных организациях, имеющих дело с многоязычным политематическим массивом документов. Таковы условия работы в Комиссии Европейских сообществ в Брюсселе, где вся документация должна появляться одновременно на девяти рабочих языках. Поскольку требования к переводу здесь высоки, МП нуждается в постредактировании.

3. В службах, осуществляющих перевод технической документации, сопровождающей экспортируемую продукцию. Переводчики не справляются в требуемые сроки с обширной документацией (так, спецификации к самолетам и другим сложным объектам могут занимать до 100 000 и более страниц). Структура и язык технической документации достаточно стандартны, что облегчает МП и даже делает его предпочтительным перед ручным переводом, так как гарантирует единый стиль всего массива. Поскольку перевод спецификаций должен быть полным и точным, продукция МП нуждается в постредактировании.

4. Для синхронного или почти синхронного перевода некоторого постоянного потока однотипных сообщений. Таков поток метеосводок в Канаде, который должен появляться одновременно на английском и французском языках. С 1975 г. задачу перевода с английского языка на французский выполняет СМП TAUM-METEO — единственная в мире полностью автоматическая СМП.

Помимо практической потребности делового мира в СМП, существуют и чисто научные стимулы к развитию МП: стабильно работающие экспериментальные системы МП являются опытным полем для проверки различных аспектов общей теории понимания, речевого общения, преобразования информации, а также для создания новых, более эффективных моделей самого МП.

История МП хорошо документирована. Кроме большого количества теоретических, отчетных и библиографических материалов, публикаций в периодической печати, в документах конференций и т. п., существует несколько фундаментальных обзорных работ. Обзоры [Bruderer, 1982; Slocum, 1985; Hutchins, 1986; Nirenburg, 1987], а также материалы международных конференций по МП [MT Summit, 1987, IC MT, 1988] наиболее полно освещают современное состояние МП.

Задачу использования ЭВМ для перевода текстов с одних языков на другие впервые в явном виде сформулировали А. Бут и У. Уивер в 1946 г. Начиная с 1949 г. в США появляется ряд научных коллективов, разрабатывающих проблематику МП: в Массачусетском технологическом институте (МТИ), в Вашингтоне, в Калифорнийском университете, в Национальном бюро стандартов в Лос-Анджелесе, в Техасском университете и др. В 1952 г. проходит первая международная конференция по МП в МТИ (США). В 1954 г. проводится знаменитый Джорджтаунский эксперимент: перевод с русского языка на английский. Хотя этот первый эксперимент по МП был очень ограничен (250 слов в словаре, тщательно отобранные фразы и т. д.), его успех обеспечил бурное развитие работ по МП в течение следующих 16 лет.

В 1955 г. проходит первый эксперимент по МП в СССР — перевод с английского языка на русский текстов из области прикладной математики (на базе словаря объемом 2300 слов); в дальнейшем эти разработки вошли в состав системы АМПАР (см. § 12.2). Начались работы по МП в Институте прикладной математики АН СССР (под руководством О. С. Кулагиной и И. А. Мельчука). Там были созданы последовательно три экспериментальные СМП: с французского языка на русский ФР-1, с английского на русский и с французского на русский ФР-П, относимые авторами соответственно к первому, полвторому и второму поколениям систем МП [Кулагина, 1979].

В 1959 г. открывается Лаборатория машинного перевода (ЛМП) в МГПИИЯ (сейчас МГИИЯ) им. М. Тореза. На материале разных языков строятся модели систем МП, разрабатывается ряд концепций, получивших «вторую жизнь» существенно позже: идея семантического языка-посредника, понятие семантических падежей и фреймов (правда, в иной терминологии), принципы единообразного представления лексических и грамматических описаний и осу-

ществления обработки текста «под управлением словаря» (см. ниже § 10.2, 10.3) и др. Эти концепции описаны в ряде публикаций¹.

Первой масштабной работой ЛМП МГИИЯ было создание лингвистического обеспечения системы англо-русского автоматического перевода (АРАП), словарь которой, включавший многоаспектную лингвистическую информацию (в том числе семантические толкования слов, релевантные для этих слов грамматические правила, сведения о лексической сочетаемости и т. п.), насчитывал примерно по 10 000 английских и русских словарных статей, а грамматика (строившаяся, в свою очередь, в виде словаря синтаксических, морфологических и семантических признаков слов, имеющего принципиально тот же формат, что и лексический словарь) — порядка 500 правил. Система не была реализована из-за отсутствия технических возможностей, но идеи ее использованы в более поздних разработках, в частности в системе ЯРАП (см. § 11.4). ЛМП участвовала в словарных работах для системы ЭТАП-1; до сих пор выполняются словарные работы для системы ФРАП (см. § 12.3, 12.4).

Теоретические и экспериментальные работы по МП в СССР велись также в Институте языкознания АН СССР, в ЦНИИПИ, в ряде коллективов Ленинграда, Тбилиси, Ташкента, Кишинева, Киева, Минска, Чимкента и др.

Работы первого периода МП полно отражены в критико-библиографическом справочнике «Автоматический перевод 1949—1963»², охватывающем 1500 публикаций по МП. Столь же фундаментальна вторая часть справочника³.

В 1966 г. специально созданный комитет Национальной академии наук США ALPAC сделал официальное заключение о нерентабельности МП по сравнению с обычным переводом. Работы в области МП пошли на убыль, но не прекратились, и к середине 70-х годов сложился тот тип СМП, который лежит в основе почти всех последующих разработок.

Возобновились работы по МП в СССР. В Москве в 1974 г. работы по МП начались в ИНФОРМЭЛЕКТРО, где созданы СМП с французского языка на русский (ЭТАП-1) и с английского на русский (ЭТАП-2). В 1974 г. открывается Всесоюзный центр переводов научно-технической литературы и документации ГКНТ и АН СССР (ВЦП), где была начата разработка трех систем промышленного масштаба: англо-русской (АМПАР), немецко-русской (НЕРПА) и французо-русской (ФРАП). Эти системы описаны в гл. 12.

С 1975 г. начинается бурное развитие СМП в Японии. К 1988 г. там работает или разрабатывается около трех десятков систем. Около 10 систем машинного перевода имеется в КНР, из них две — промышленные. В 1987 г. в Японии состоялась конференция по вопросам промышленной жизни МП [MT Summit, 1987]. В 1988 г. в Будапеште проходит международная конференция по МП [IC MT, 1988] как часть конференции по вычислительной лингвистике [COLING, 1988], в программу которой тоже вошли доклады по МП. Широкий спектр проблем МП рассматривается в журнале *Computers and Translation*, см., например, [CAT, 1988].

10.2. Периодизация и классификация систем машинного перевода

Системы машинного перевода можно классифицировать по нескольким основаниям. Одно из них — принятый в системе тип лингвистической стратегии. С этой точки зрения выделяют четыре периода, каждый из которых характеризуется преимущественным развитием систем того или иного типа.

¹ Машинный перевод: Сб статей. — М.: ИТМ и ВТ АН СССР, 1961; Машинный перевод и прикладная лингвистика: Сб. статей. — М.: МГПИИЯ. 1964. — Вып. 8; 1968. — Вып. 11; 1974. — Вып. 17.

² Мельчук И. А., Равич Р. Д. Автоматический перевод (1949—1963): Критико-библиографический справочник. — М., 1967, — 517 с.

³ Melčuk J. A., Ravič P. D. *Traduction Automatique, 1964—1970. — Automatic Translation. Montréal, 1978. — 421 p.*

Начальный период «бурного развития» (конец 40-х — середина 60-х годов) характеризуется преимущественным развитием *прямых систем МП*, реализующих лобовое решение проблемы перевода и обеспечивающих результаты, близкие к пословному переводу (так называемые системы *первого поколения*).

Второй период (середина 60-х — середина 70-х годов) отмечен интенсивным развитием синтаксических теорий и разработкой на их основе СМП *второго поколения*.

Третий период (середина 70-х — середина 80-х годов) можно назвать периодом экстенсивного развития СМП: они получили промышленную жизнь. Техника морфологического и синтаксического анализа хорошо освоена, на повестке дня — семантика. Но ожидаемого выхода к СМП *третьего поколения*, осуществляющего перевод через семантические структуры, не произошло. В качестве компенсации получают широкое развитие интерактивные СМП, комбинирующие труд человека и ЭВМ. Другим внешним решением семантических трудностей является ориентация на перевод ограниченных классов текстов, в частности, настроенных на узкую предметную область (ПО).

Четвертый период (со второй половины 80-х годов) характеризуется резким возрастанием интереса к МП как в практическом, так и в теоретическом плане: МП — сложная область, на которой отрабатываются новые информационные технологии. Появляются все больше многоязычных систем. Большие надежды возлагаются на мощные лексические и терминологические базы данных, базы знаний. Сближается проблематика МП и ИИ широкого профиля: в МП привлекаются семантические теории, созданные для экспертных и других систем ИИ.

Таким образом, СМП первого и второго периодов были противопоставлены прежде всего по типу лингвистической стратегии. В системах первого и полуторного поколений операция перевода требует минимума преобразований: *исходный текст* постепенно превращается в текст на *выходном языке* путем замены всех его элементов, найденных в словаре, на переводные эквиваленты, никакая языковая модель не требуется, кроме *переводных* (в основном лексических и позиционных) *соответствий*. Учитывается лишь локальный контекст, он же позволяет собирать некоторые сложные единицы — *обороты* (поэтому такой перевод называют *пословным*, или *пословно-пооборотным*). Для систем данного типа характерны *бинарность*, отсутствие промежуточных структур (анализ ведется сразу в категориях выходного языка), *одновариантность*. Их сильная сторона — простота устройства и вытекающая отсюда большая скорость работы.

В системах второго поколения переводные соответствия устанавливаются не «прямым» способом, а только после того, как для каждого предложения выявлена в результате анализа его *синтаксическая* или синтактико-семантическая *структура* (или несколько альтернативных вариантов такой структуры). *Анализ* и *синтез* независимы (анализ, как правило, *многовариантный*, ведется в категориях входного языка, синтез — в категориях выходного), так что связь того и другого этапов обеспечивается введением особого этапа *межъязыковых операций* (собственно перевода, *трансфера*).

На этом лингвистические основания классификации СМП прервались, так как перевод через *семантический язык-посредник* (ЯП), универсальный для разных пар естественных языков, не был обеспечен единой общепризнанной лингвистической теорией. Определение и развитие СМП *третьего* и *четвертого* поколений остается делом будущего. Системы *пятого поколения* выделены в японском проекте по другому основанию: они должны стать частью компьютеров пятого поколения. Это развитые многоязычные системы, базирующиеся на самой передовой информационной технологии. Их лингвистические возможности предполагается расширить освоением речевого ввода и вывода.

В литературе 80-х годов больше говорится о делении СМП не на поколения, а на системы с *прямым* и *непрямым переводом*, а последних — на системы с *трансфером* и с *языком-посредником*. Иногда СМП классифицируют на *синтаксически-ориентированные* и *лексически-ориентированные* (или СМП, ра-

ботающие *под управлением словаря*, англ. *lexicon-driven systems*, — они приближаются по технике анализа к системам класса ИИ). Дополнительная характеристика компонентов этих трех типов систем содержится в § 10.3.

В 80-е годы в отдельный класс выделяют СМП, основанные на знаниях (*knowledge-based systems*). В системах этого класса (представляющего собой подкласс систем ИИ) в качестве отдельного компонента включаются *экстралингвистические знания* (знания о ПО), хотя они могут иметь те же формы представления, что и собственно лингвистическая информация (т. е. записываться в словарях и грамматиках). Отчетливо к этому классу принадлежат те СМП, которые используют при анализе *концептуальную сеть знаний*.

Логико-алгоритмические основы современных больших наиболее развитых СМП (GETA, METAL, EUROTRA и др.) обычно настолько гибки, что могут включать любую лингвистическую теорию и допускают разные их комбинации. Однако они имеют, как правило, собственную модель процесса перевода, что позволяет говорить о системах типа SYSTRAN, METAL и т. д.

Следующие два основания классификации примыкают к лингвистическому.

По количеству привлекаемых языковых пар СМП делятся на *двуязычные* (реализующие функцию перевода только для данной языковой пары) и *многоязычные*. Те и другие в зависимости от техники лингвистического анализа могут быть либо *бинарными* (если анализ входного языка ведется в категориях выходного), либо *универсальными* (если устройство анализа не зависит от выходного языка). Универсальная двуязычная СМП может легко стать многоязычной при комбинации с компонентами анализа и/или синтеза других универсальных систем (такова, например, система EUROTRA; см. также в § 11.3 о системе SEMSYN, комбинирующей компонент синтеза, разработанный для немецкого языка, с компонентами анализа и межязыковых операций, разработанными независимо от него в японско-английской системе ATLAS-II). Система же, состоящая из совокупности бинарных СМП, может быть названа *многоязычной*, но не является универсальной (системы SYSTRAN, АНРАП).

По тематической ориентации различают системы *монотематические*, построенные на одну ПО (таких большинство: TAUM-METEO, SPANAM, METAL, TITUS), и *политематические*. Иногда СМП имеют ограничения на структуру вводимых текстов (СМП с *ограниченным ЕЯ*, например, система TITUS или TITRAN, переводящая только заголовки).

Классификация СМП может учитывать также технологические характеристики: масштабность, степень реализованности, долю участия человека в процессе МП. С точки зрения соотношения машина-человек можно говорить о *полностью автоматическом переводе* и о *человекомашинном переводе* (ЧМП). Системы, осуществляющие полностью автоматический перевод, — это СМП большого масштаба, реализованные на больших ЭВМ. Они выполняют перевод без участия человека, что не исключает ни некоторой предварительной обработки текста, ни постредактирования (как это делается и в случае обычного перевода). Но за сам процесс перевода — от ввода исходного до получения конечного текста — «ответственная» СМП, располагающая словарями, грамматикой и программами. Системы типа ЧМП выполняют перевод в интерактивном режиме, обычно на малых — мини- или микро- (персональных) ЭВМ (иногда соединенных с основной, центральной ЭВМ). Выделяют две группы таких систем: типа НАМТ (англ. *human-aided machine translation*) — МП с участием человека и типа МАНТ (англ. *machine-aided human translation*) — человеческий перевод с участием ЭВМ. В системах первой группы перевод выполняет СМП, обращаясь к человеку за решением сложных случаев (снятие неоднозначности, выбор одного варианта из нескольких и т. п.). В системах второй группы за выполнение перевода отвечает человек, который работает за дисплеем в интерактивном режиме и может обращаться к ЭВМ, например, для поиска слов в машинных словарях и конкордансах, а также в удаленных банках терминов. Предредактирование при этом, как правило, не нужно (ср., впрочем, подход, реализуемый в системе TAS-1, § 11.3), а постредактирование осуществляется, как при обычном переводе.

Поиск оптимальных соотношений человек-машина продолжается, но интерактивные СМП уже вышли на коммерческую основу (Weidner, ALPS и др.).

С последней разновидностью СМП смыкаются автоматические словари (АС) и терминологические банки данных (ТБД), являющиеся вспомогательным средством при переводе. Обращение к ним обычно предшествует процессу перевода. Они могут выдавать глоссарии по определенной ПО, лексические единицы по запросам и т. д. Системы АС, ТБД и типа МАНТ (в помощь переводчику) здесь не рассматриваются; описываются лишь СМП, всю или основную работу в которых выполняет ЭВМ.

С точки зрения масштаба и степени разработанности СМП можно разбить на три основных класса: *промышленные, развивающиеся и экспериментальные*. Определения этих классов СМП даются в гл. 11 (где данная классификация принята за основу при описании современных систем МП).

10.3. Лингвистическое обеспечение систем машинного перевода

Процесс МП представляет собой последовательность преобразований, применяемых к входному тексту и превращающих его в текст на выходном языке, который должен максимально воссоздавать смысл и, как правило, структуру исходного текста, но уже средствами выходного языка. К лингвистическому обеспечению СМП относится весь комплекс собственно лингвистических, металингвистических и так называемых «экстралингвистических» знаний, которые используются при таком преобразовании.

В классических СМП, осуществляющих не прямой перевод по отдельным предложениям (пофразный перевод), каждое предложение проходит последовательность преобразований, состоящую из трех частей (этапов): *АНАЛИЗ* → *ТРАНСФЕР* (*межъязыковые операции*) → *СИНТЕЗ*. В свою очередь, каждый из этих этапов представляет собой достаточно сложную систему промежуточных преобразований.

Цель этапа анализа — построить структурное описание (*промежуточное представление, внутреннее представление*) входного предложения. Задача этапа трансфера (собственно перевода) — преобразовать структуру входного предложения во внутреннюю структуру выходного предложения. К этому этапу относятся и замены лексем входного языка их переводными эквивалентами (лексические межъязыковые преобразования). Цель этапа синтеза — на основе полученной в результате анализа структуры построить правильное предложение выходного языка.

Лингвистическое обеспечение стандартной современной СМП включает: 1) *словари*; 2) *грамматики*; 3) формализованные *промежуточные представления* единиц анализа на разных этапах преобразований. Помимо стандартных, в отдельных СМП могут иметься и некоторые нестандартные компоненты. Так, экспертные знания о ПО могут задаваться с помощью специальных *концептуальных сетей*, а не в виде словарей и грамматик.

Механизмы (алгоритмы, процедуры) оперирования с имеющимися словарями, грамматиками и структурными представлениями относят к математико-алгоритмическому обеспечению СМП.

Одно из необходимых требований к современным СМП — высокая модульность. С лингвистически содержательной точки зрения это означает, что анализ и следующие за ним процессы строятся с учетом теории лингвистических уровней. В практике создания СМП различают такие уровни анализа:

досинтаксический анализ (в него входит *морфологический анализ* — МорфАн, анализ оборотов, неопознанных элементов текста и др.);

синтаксический анализ СинАн (строит *синтаксическое представление* предложения, или СинП); в его пределах может выделяться ряд подуровней, обеспечивающих анализ разных типов синтаксических единиц;

семантический анализ СемАн, или логико-семантический анализ (строит аргументно-предикатную структуру высказываний или другой вид *семантического представления* предложения и текста);

концептуальный анализ (анализ в терминах концептуальных структур, отражающих семантику ПО). Этот уровень анализа используется в СМП, ориентированных на очень ограниченные ПО. По сути дела, концептуальная структура является проекцией схем ПО на лингвистические структуры, часто даже не на семантические, а на синтаксические. Только для очень узких ПО и ограниченных классов текстов концептуальная структура совпадает с семантической; в общем случае полного совпадения не должно быть, так как текст подробнее любых концептуальных схем.

Синтез (см. п. 10.3.4) теоретически проходит те же уровни, что и анализ, но в обратном направлении. В работающих системах обычно реализован только путь от СинП до цепочки слов выходного предложения.

Лингвистическое разграничение разных уровней может проявляться также в разграничении используемых в соответствующих описаниях формальных средств (набор этих средств задается для каждого уровня отдельно). На практике часто задаются отдельно лингвистические средства МорфАн и совмещаются средства СинАн и СемАн. Но разграничение уровней может оставаться только содержательным при использовании в их описаниях единого формализма, пригодного для представления информации всех выделяемых уровней (таков, например, формализм Q-систем).

С технической точки зрения модульность лингвистического обеспечения означает отделение структурного представления фраз и текстов (как текущих, временных знаний (о тексте) от «постоянных» знаний о языке, а также языковых знаний — от знаний ПО; отделение словарей от грамматик, грамматик — от алгоритмов их обработки, алгоритмов — от программ. Провозглашенный в СМП второго поколения принцип отделения лингвистических знаний от механизмов изначально понимался в том смысле, что словари и грамматики должны задаваться декларативно, статически (и формулироваться лингвистами), тогда как конкретные процедуры их использования (механизмы) должны создаваться в основном программистами. Однако возможно процедурное задание грамматик и даже словарей, сближение словарей и грамматик по формату записи, по способу организации и типам включаемых в них сведений и т. д. (см. [Программное, 1984], а также п. 10.3.2), что отнюдь не означает отказа от модульной организации. Смысл последней состоит в том, что изменения внутри модуля не влияют на вид информации, поступающей на его вход и выдаваемой на его выходе, так что отдельные компоненты грамматик и словарей можно менять и дополнять, не меняя всей системы.

Конкретные соотношения различных модулей системы (словари — грамматики, грамматики — алгоритмы, алгоритмы — программы, декларативные — процедурные знания и др.), включая распределение лингвистических данных по уровням, — это то основное, что определяет специфику СМП.

10.3.1. Словари. Словари анализа, как правило, одноязычные. Они должны содержать всю информацию, необходимую для включения данной лексической единицы (ЛЕ) в структурное представление. Часто разделяют словари основ (с *морфолого-синтаксической информацией*: часть речи, тип словоизменения, подкласс, характеризующий синтаксическое поведение ЛЕ и т. п.) и словари *словозначений*, содержащие *семантическую* и *концептуальную информацию*: семантический класс ЛЕ, *семантические надежды* (валентности), условия их реализации во фразе и т. д.

Во многих системах разделены словари *общеупотребительной* и *терминологической лексики*. Такое разделение дает возможность при переходе к текстам другой предметной области ограничиваться лишь сменой терминологических словарей. Словари сложных ЛЕ (оборотов, конструкций) образуют обычно отдельный массив, словарная информация в них указывает на способ «собираения» такой единицы при анализе. Часть словарной информации может задаваться в процедурной форме, например, *многозначным словам* могут сопоставляться алгоритмы *разрешения* соответствующего типа *неоднозначности* (см. § 12.2 о системе АМПАР). Новые виды организации словарной информации для целей МП предлагают так называемые «лексические базы знаний» (см. описа-

ние системы DLT в [Paragaaij, 1986]). Наличие разнородной информации о слове (называемой лексическим универсумом слова) приближает такой словарь скорее к энциклопедии, чем к традиционным лингвистическим словарям.

10.3.2. Грамматики и алгоритмы. Грамматика и словарь задают лингвистическую модель, образуя основную часть лингвистических данных. Алгоритмы их обработки, т. е. соотношения с текстовыми единицами, относят к *математико-алгоритмическому обеспечению* системы.

Разделение грамматик и алгоритмов важно в практическом смысле тем, что позволяет менять правила грамматик, не меняя алгоритмов (и соответственно программ), работающих с грамматиками. Но далеко не всегда такое разделение возможно. Так, для системы с *процедурным заданием грамматики* и тем более с процедурным представлением словарной информации такое разделение нерелевантно. Алгоритмы принятия решений в случае недостаточной (неполнота входных данных) или избыточной (вариантность анализа) информации в большой мере эмпиричны, их формулировка требует лингвистической интуиции. Задание общего управляющего алгоритма, ведающего порядком вызова разных грамматик (если их несколько в одной системе), также требует лингвистического обоснования. Тем не менее существующая тенденция — отделить грамматики от алгоритмов так, чтобы все лингвистически содержательные сведения задавались в статической форме грамматик, а алгоритмы сделать настолько абстрактными, что они смогут вызывать и обрабатывать разные лингвистические модели.

Наиболее четко разделение грамматик и алгоритмов наблюдается в системах, работающих с *контекстно-свободными (КС) грамматиками (КСГ)*, где модель языка — *грамматика с конечным числом состояний*, а алгоритм должен обеспечить для произвольно взятого предложения дерево его вывода по правилам грамматики, и если таких выводов несколько, то перечислить их. Такой алгоритм, представляющий собой формальную (в математическом смысле) систему, называется *анализатором*. Типы анализаторов, используемых в системах машинного перевода, описаны в [Програмное, 1984]. Описание грамматики служит для анализатора, обладающего универсальностью, таким же входом, как и анализируемое предложение. Анализаторы строятся для классов грамматик, хотя учет специфических особенностей грамматики может повысить эффективность анализатора.

Граматики *синтаксического уровня* — наиболее разработанная часть и с точки зрения лингвистики, и с точки зрения их обеспечения формализмами. Укажем основные типы грамматик и реализующих их алгоритмов (в литературе по МП их часто описывают как одну совокупность).

Цепочечная грамматика фиксирует порядок следования элементов, т. е. линейные структуры предложения, задавая их в терминах грамматических классов слов (артикл+существительное+предлог...) или в терминах функциональных элементов (подлежащее+сказуемое). Примером реализации такой языковой модели является *предсказуемый синтаксический анализ*: идентифицируемая грамматическая категория слова предсказывает (с определенной долей вероятности) появление грамматической категории следующего за ним слова. Стратегия анализа — «слева направо»: перебор слов, проверка предсказаний, их изменение и добавление новых предсказаний регулируются механизмом «магазинной памяти» (last in first out). Формально, это реализация на ЭВМ грамматик с конечным числом состояний.

Грамматика составляющих (или грамматика непосредственно составляющих — НСГ) фиксирует лингвистическую информацию о группировке грамматических элементов, например именная группа (состоит из существительного, артикля, прилагательного и других модификаторов), предложная группа (состоит из предлога и именной группы) и т. д. до уровня предложения. Грамматика строится как набор *правил подстановки*, или *исчисление продукций* вида $A \rightarrow B \dots$. С. НСГ представляют собой грамматики *порождающего типа* и могут использоваться как при анализе, так и при синтезе: предложения языка порождаются многократным применением таких правил.

Грамматика зависимостей (ГЗ) задает иерархию отношений элементов предложения (главное слово определяет форму зависимых). Анализатор в ГЗ основан на идентификации *хозяев* и их зависимых (*слуги*). Главным в предложении является глагол в личной форме, так как он определяет число и характер зависимых существительных. Стратегия анализа в ГЗ — *сверху вниз* (top-down): сначала идентифицируются хозяева, затем слуги, или *снизу вверх*: (bottom-up): хозяева определяются процессом подстановки.

Категориальная грамматика Бар-Хиллела — это версия грамматики составляющих, в ней только две категории — предложения $\bar{S}$ и имени p . Остальные определяются в терминах способности комбинироваться с этими главными в структуре НС. Так, переходный глагол определен как $p \setminus S$, так как он сочетается с именем p слева от него, образуя предложение S .

Все ГЗ и НС-грамматики формально эквивалентны и принадлежат классу КСГ. Для КС-языков существует широкий выбор методик построения анализаторов. Однако чистые КСГ недостаточны для описания естественного языка. Н. Хомский предложил дополнить КСГ (*базовый компонент* системы описания языка) набором *трансформационных правил*, работающих с *деревьями составляющих*. Хотя правила такого рода обычно присутствуют в системах МП, трансформационные грамматики в смысле Хомского не получили распространения в системах обработки текста. Другая возможность — использование *контекстно-зависимых правил*. Именно этот подход применяется сейчас в той или иной форме во всех моделях, использующих понятие грамматик структуры составляющих [Программное, 1984].

Существует много способов учета контекстных условий: *грамматики метаморфозы* и их варианты, *ГОС*, *РСР*, *ЛФГ* и др. (см. ниже). Все они являются *расширениями КС-правил*. В общем виде это значит, что правила продукций переписываются так: $A[a] \rightarrow B[b] \dots C[c]$, где малыми буквами обозначены условия, тесты, инструкции и т. д., расширяющие исходные жесткие правила и дающие грамматике гибкость и эффективность.

В грамматике *обобщенных составляющих* — ГОС (generalized phrase structure grammar) [Gazdar et al., 1985] введены метаправила, являющиеся обобщением закономерностей правил КСГ.

В грамматиках *расширенных сетей переходов* — РСР (augmented network transition grammar) предусмотрены тесты и условия к дугам, а также инструкции, которые надо выполнить в случае, если анализ пошел по данной дуге. В разных модификациях РСР дугам может присписываться вес, тогда анализатор может выбирать путь с наибольшим весом. Условия могут разбиваться на две части: контекстно-свободные и контекстно-зависимые.

Разновидностью РСРГ являются *каскадные РСРГ*. *Каскад* — это РСР, снабженная действием transmit. Это действие вызывает остановку процесса в данном каскаде, запоминание информации о текущей конфигурации в стеке и переход к более глубокому каскаду с последующим возвратом в исходное состояние. РСР обладает рядом возможностей трансформационных грамматик. Она может использоваться и как генерирующая система.

Метод анализа с помощью *граф-схемы* (chart analysis) позволяет сохранить частичные результаты и представить варианты анализа.

Новым и сразу завоевавшим популярность методом грамматического описания является *лексико-функциональная грамматика* (ЛФГ) [Bresnan et al., 1982]. Она устраняет необходимость трансформационных правил. Хотя ЛФГ основывается на КСГ, проверочные условия в ней отделены от правил подстановки и «решаются» как автономные уравнения.

Унификационные грамматики (УГ) представляют собой следующий после граф-схем этап обобщения модели анализа: они способны воплощать грамматики различных видов. УГ содержит четыре компонента: пакет унификаций, интерпретатор для правил и лексических описаний, программы обработки направленных графов, анализатор с помощью граф-схемы. УГ объединяют грамматические правила со словарными описаниями, синтаксические валентности с семантическими.

Центральной проблемой любой системы анализа ЕЯ является проблема выбора вариантов. Для ее решения грамматики синтаксического уровня дополняются вспомогательными грамматиками и методами разбора сложных ситуаций. В НС-грамматиках применяют *фильтровый* и *эвристический методы*. Фильтровый метод состоит в том, что сначала получают все варианты анализа предложения, а затем отбраковывают те, которые не удовлетворяют некоторой системе условий-фильтров. Эвристический метод с самого начала строит лишь часть вариантов, более правдоподобных с точки зрения заданных критериев (ср. с методом «первый — лучший»). Использование весов для отбора вариантов является примером применения эвристических методов в анализе.

Семантический уровень гораздо меньше обеспечен теорией и практически разработками. Традиционной задачей семантики считается снятие *неоднозначности* синтаксического анализа — структурной и лексической. Для этого используется аппарат *селективных ограничений*, который привязан к рамкам предложений, т. е. вписывается в синтаксическую модель. Альтернативные подходы развивались на ранних этапах МП — это *тезаурусный подход* М. Мастерман (Великобритания) и *корреляционный анализ* С. Чеккато (Италия).

Системы СЕТА и LRC развивали *логико-семантический анализ*, представляющий структуру предложений в терминах логических отношений (предикаты, аргументы, атрибуты), уменьшающих расстояние между *поверхностными структурами* разных языков. Впоследствии эти системы перешли к трансферной стратегии. К аргументно-предикатным структурам близки *глубинно-синтаксические структуры*.

Наиболее распространенный тип СемАн основан на так называемых *падежных грамматиках*, получивших популярность после появления известной статьи Ч. Филлмора в 1968 г. В основе грамматики — понятие *глубинного*, или *семантического*, *падежа*. *Падежная рамка* глагола является расширением понятия *валентности*: это набор смысловых отношений, которые могут (обязательно или факультативно) сопровождать глагол и его вариации в тексте. Примеры *глубинных падежей*: агент, адресат, цель и другие, — они универсальны и языково-независимы. В пределах одного языка один и тот же *глубинный падеж* реализуется разными *поверхностными предложно-падежными формами*. *Глубинные падежи* в принципе позволяют выходить за рамки предложения, а выход в текст означает переход к семантическому уровню анализа.

Грамматика семантического уровня в чистом виде (вне связи с синтаксисом) представлена в семантическом анализаторе И. Уилкса. Его модель *семантической предпочтительности* включает такие инструменты собственно СемАн, как тезаурусная информация, правила вывода «по здравому смыслу», понятие псевдотекста, декомпозицию лексем на семантические примитивы, что позволяет доказывать *семантическую связность* даже на трудном лингвистическом материале (метафорические словоупотребления).

Другая несинтаксическая грамматика — модель *концептуальных зависимостей* Р. Шенка в системе, основанной на знаниях в ограниченной ПО. Понятийные сети, концептуальные сети, фреймы, схемы ПО используются как грамматика концептуального уровня.

Поскольку семантическая информация в отличие от синтаксической, опирающейся в первую очередь на грамматику, сосредоточена в основном в словарях, в 80-е годы интенсивно разрабатываются грамматики, позволяющие «лексикализовать» КСГ. Это ЛФГ, грамматика объединения деревьев (TAG-tree adjoining grammar) и др. Ведется разработка грамматик, основанных на исследовании свойств дискурса [COLING, 1988].

10.3.3. Типы промежуточных представлений. В системах, претендующих на надежную и содержательно обоснованную переводческую обработку текстов, в качестве отдельного компонента лингвистического обеспечения выделяются *временные знания* об обрабатываемом фрагменте текста (в отличие от *постоянных, долговременных знаний*, содержащихся в словарях и грамматиках). В основном это сведения о *внутренней структуре* такого фрагмента (синтаксической, семантической, анафорической и т. п.) — его *формализованное представление*,

выражаемое в терминах составляющих, зависимостей, семантических сетей, фреймов и т. д. При анализе структурное представление используется для контроля за правильностью разбора предложения, а также является основой для перевода (входом в этап перевода и затем синтеза).

Можно выделить пять типов промежуточных структур.

1. Собственно СинП в разных вариантах: в терминах дерева НС или ГЗ; есть гибридные СинП, сочетающие признаки НС и ГЗ. Если СинАн дает несколько вариантов разбора, они могут быть представлены либо несколькими деревьями, либо одним *обобщающим деревом*. Выделяют *нормализованную структуру* (ТАУМ), *каноническую структуру* (КВМТ) и др. Назначение этих типов СинП — максимально подготовить структуру к переводу, освободив от специфических для входного языка явлений. Существуют также *поверхностное* (использующее до 100 типов отношений) и *глубинное* (6—7 отношений) СинП, *комбинированное СинП* (объединяющее поверхностные и глубинные характеристики). Они отличаются друг от друга характером узлов и связей. Единицы СинП могут принадлежать *функциональному синтаксису* (от 5 до 20 функциональных синтаксических отношений и указание роли при узлах). В GETA — *многоуровневое СинП* (узлам приписана информация всех уровней анализа).

2. СинСемП — представление на синтаксической основе, но в терминах *глубинных семантических отношений* (переход к структуре выходного предложения требует только смены лексики).

3. СемП — представление на семантической основе, безразличное к членению текста на предложения.

4. Концептуальная структура в терминах единиц из ограниченной ПО.

5. Язык-посредник (ЯП), или *интерлингва*.

Можно считать, что между этими представлениями существует отношение не альтернативы, а развития: каждое следующее «глубже» предшествующих. Первые две структуры имеют лингвистическую основу, третья и четвертая — понятийную, пятая базируется на единицах, не принадлежащих ни входному, ни выходному языку.

10.3.4. Лингвистическое обеспечение этапов трансфера и синтеза. Синтез в стандартных СМП трехчастного типа устроен проще, чем анализ, если он обеспечивается соответствующим входом, т. е. если после работы трансфера построено одно правильное дерево фразы. Используется та же лингвистическая модель, что и в анализе, в виде *порождающей грамматики*, программа синтеза работает как *детерминированный преобразователь*: необходимые для построения выходной словоформы грамматические категории определяются свойствами подчиняющего узла в дереве предложения. Одноязычные словари синтеза содержат всю информацию, которая позволяет *синтезировать* требуемую словоформу; в случае, если таковая не может быть построена, предлагаются замены (синтаксические или лексические дериваты).

Одна из задач синтеза — *линеаризация узлов дерева*, определяющая нужный порядок слов в синтезируемом предложении. Стандартным решением вопроса о порядке слов является передача на этап синтеза информации о взаимном расположении слов (или групп, конструкций) в исходном предложении с учетом тех регулярных изменений, которые производятся на этапе трансфера (например, изменения порядка следования компонентов группы «существительное + прилагательное» при французско-русском переводе). Более тонкое решение этой проблемы требует учета таких категорий, как темо-ремное членение, модально-временная рамка, семантика артиклевых употреблений и других явлений семантического уровня.

Правильность дерева, построенного на этапе трансфера, или выбор одного из нескольких вариантов, если они остались, проверяется теми же методами, что и на этапе анализа.

Таким образом, решение всех содержательных вопросов, которые могут повлиять на построение хорошего выходного предложения, остаются в компетенции этапов анализа и трансфера (межязыковых операций).

Компонент трансфера задается двумя частями:

словарь трансфера, или *переводной* (межъязыковой) *словарь*,—это двуязычный словарь лексических замен. Как правило, он одностороннен: ЛЕ (словозначение) входного языка→ЛЕ выходного языка, поскольку для каждого слова (с нерасчлененными значениями) существует несколько переводов в зависимости от контекста; возможно, впрочем, и *обратимые переводные словари* (см. § 11.4 о системе NTT-CSK);

словарь *структурных соответствий*, или *грамматика трансфера*.

Работа алгоритма преобразования состоит в сравнении структур (pattern matching), вычислении нужных значений и замене структуры входного предложения структурой выходного. Эти две части связаны, так что собственно перевод состоит иногда в замене целых поддеревьев (слова с его структурным контекстом) их лексическими и структурными эквивалентами.

Для некоторых языковых пар структурное и лексическое соответствия достаточно прямые (переводится слово в слово, конструкция—в конструкцию), но и при этом степень неоднозначности перевода отдельных слов высока. В случае очень разных языков (японский vs. любой из европейских) грамматика трансфера сложнее. Например, в проекте Ми она включает два дополнительных подэтапа: преобразование внутреннего представления входного предложения в дотрансферное→собственно трансфер→преобразование после трансфера; лишь затем работает грамматика синтеза. Один из способов задания грамматики трансфера—*контрастивная грамматика*, представляющая собой собрание грамматических явлений, поверхностное выражение которых в данной паре языков различно.

Один из трудных вопросов этапа трансфера—определение порядка применения правил. Чтобы разгрузить этот громоздкий этап, можно либо строить более глубокие структуры при анализе, позволяющие раньше выявлять и снимать смысловую неоднозначность, либо часть задач передать синтезу, который должен будет выбирать одно из нескольких деревьев.

10.4. Математическое и программное обеспечение машинного перевода

К математическому и программному обеспечению СМП относятся: *формализмы*, используемые для задания лингвистических данных (словарей и грамматик); специальные (МП-ориентированные) *алгоритмические языки*; *языки программирования*.

Языки задания грамматик могут рассматриваться как языки программирования очень высокого уровня, допускающие сложные типы данных и ассоциированные операторы. Метаязыки описания данных обычно одинаковы для входного и выходного языков. Впрочем, при работе с разнотипными языками (например, с парами японский vs. европейский язык) они могут быть и различны (для японского языка обычно используются грамматики зависимостей—чаще всего падежные, для европейского, особенно для английского,—НС-грамматики).

Формальными языками являются также *языки описания словарей*. Высоким уровнем формализованности (при большой глубине смысловой информации) отличаются словари отечественных систем АРАП, ЭТАП, ФРАП. Еще более высокий уровень организации всех словарных сведений предполагается достичь в составе Лексического банка знаний системы DLT. Такие словари можно называть словарями-грамматиками.

Алгоритмический язык (АЯ) характеризуется прежде всего классом алгоритмов, которые можно описать его средствами. Класс, в свою очередь, характеризуется объектами, с которыми работают алгоритмы. Поэтому при описании АЯ, ориентированных на МП, следует прежде всего указать объекты соответствующих алгоритмов. Так, операторный язык, используемый в СМП «прямого» типа АНРАП [Тихомиров, 1984], работает с двумя объектами: с текстом и с массивом информационный ячеек, имеющих одномерную структуру. Интересным примером АЯ, работающего с синтаксическим представлением предложения, является система ROBRA, входящая в состав программного комплекса

ARIANE-78, в основе которого лежит специальный математический аппарат. ROBRA по результату морфологического анализа (имеющего вид дерева с этикетками в узлах) осуществляет многоуровневый анализ (порождение промежуточной структуры) и участвует в генерации промежуточной и поверхностной структур выходного предложения. Показано [Программное, 1984], что на языке системы ROBRA можно писать недетерминированные анализаторы для КСГ и контекстно-зависимых грамматик.

Алгоритмический язык системы ЭТАП-2 — это развитый формальный язык для записи лингвистической информации, в котором используются предикаты и кванторы; с этого языка осуществляется трансляция на ПЛ/1 [Цинман, 1986].

При использовании языка-посредника (семантического представления) АЯ должен работать с графами (СемП — граф, не обязательно являющийся деревом). Примером служит язык ОЛЯ (Операторный Язык для Лингвистов) в рамках системы ФРАП [Машинный, 1987]. Здесь объекты оперирования — узлы графа, имеющие многоуровневую структуру.

У. Хатчинз утверждает [Hutchins, 1986], что существует большая сходимость между формализмами унификационных грамматик и нетрансформационными теориями, особенно ЛФГ, ГОС и грамматикой Монтегю. ЛФГ внедрена во многих проектах, не только экспериментальных, но и коммерческих (системы WEIDNER); ГОС предпочитается в экспериментальных системах, например в NASEV и NTPANS; грамматика Монтегю внедрена в системах ROSETTA, LUTE. Для этих формализмов наиболее подходящим оказался язык программирования Пролог — он постепенно вытесняет имевший большую популярность в системах обработки ЕЯ язык Лисп. Хотя Пролог принят как основной язык программирования для японского проекта систем пятого поколения, сейчас он используется в относительно небольшом круге систем — разработчики коммерческих систем предпочитают язык Си.

Основное программное обеспечение СМП дополняется вспомогательными программными комплексами, периферийными с точки зрения самого МП, — так называемыми front-end processors. К ним относятся средства, выполняющие следующие процедуры: 1) ввод данных; 2) коррекцию ошибок; 3) управление данными; 4) редактирование текста; 5) вывод текста; 6) коммуникацию (передачу данных); 7) пополнение и модификацию словарей и грамматик. Все эти средства обычно имеются в распоряжении переводчика и постредактора на *рабочих станциях* или *автоматизированных рабочих местах* (АРМ). Они во многом совпадают с программным обеспечением других типов автоматизированных информационных систем, поэтому здесь не описываются. Следует только упомянуть системы управления базами данных (СУБД), которые могут использоваться в СМП для создания и ведения сложных словарных массивов и баз знаний.

В последнее время стоит вопрос о внедрении многоязычных СМП с *распределенной обработкой*: анализ в пункте ввода, синтез в пункте вывода (такова, например, система DLT). Для них первостепенное значение имеют вопросы коммуникации.

10.5. Оценка систем машинного перевода

Полностью автоматический высококачественный перевод оказался практически недостижимым, как и предсказывал И. Бар-Хиллел. Конкретных пользователей промышленного МП интересует, может ли МП производить продукт, приемлемый по качеству с точки зрения некоторой конкретной задачи, является ли МП экономически эффективным (по скорости, стоимости, затратам на расширение системы и т. д.). Отказ от промышленного внедрения системы TAUM-AVIATION [Hutchins, 1986] в Канаде свидетельствует, например, о том, насколько серьезны эти экономические требования.

Характерно, что в сообщениях о коммерческих СМП чаще всего даются только рекламные примеры переводов без статистических оценок и без эксплицитного анализа качества приводимых переводов.

В явном виде вопрос о качестве переводов, обеспечиваемых автоматическими системами, о критериях и методах оценки качества ставится в экспериментальных и исследовательских СМП.

Основные параметры качества, по которым производилась оценка во многих системах, — это *понятность* (в какой мере выходной текст АП может быть понят читателем без обращения к оригиналу) и *адекватность* (насколько точно он передает смысл оригинала). В японско-английской системе Му [Нагао и др., 1986] понятность оценивается по пятибалльной шкале, адекватность — по шестибалльной. При пофразной оценке переводов, выполненных системой GETA (на массиве объемом 15 000 слов), переведенные фразы разбивались на три класса: вполне понятные — таких было 50 %, не совсем понятные — 28 % и совсем непонятные — 22 %. Для текстов общим объемом 300 000 слов результаты разбивались на два класса: понятные — 61 %, непонятные — 29 % фраз [Кулагина, 1979]. Само разбиение на классы качества достаточно субъективно.

В оценке системы ФР-2 проводится разбиение на категории самих экспертов, осуществляющих эту оценку (лингвисты, математики, смешанные группы), и учитывается три параметра: понятность, адекватность, грамматическая правильность. Оценка каждого параметра может принимать три значения: хорошо, средне, плохо. Оценка усложняется, если система дает несколько переводов одной фразы. Данные оценок приводятся в [Кулагина, 1979, с. 247].

Ряд авторов на основании своих оценок делают вывод, что, если машина дает правильно построенный и легко понимаемый перевод, он оказывается обычно верным и с точки зрения передачи смысла. Однако данные М. Нагао [Нагао и др., 1986] противоречат этому: приводятся примеры, которые получили по параметру адекватности низкую оценку, а по параметру понятности высокую.

Возможна и внутрисистемная оценка — оценка самой системой результатов каждого этапа. Самым очевидным случаем самооценки является принцип «все или ничего», т. е. принятие или непринятие следующим этапом результатов предыдущего. Этим жестким принципом долгое время руководствовались классические СМП второго поколения: если строится дерево предложения, оно поступает на синтез; если же хотя бы на одном участке неудача, вся фраза отбраковывается. В дальнейшем этот принцип был смягчен использованием эвристик, приписыванием весов, выходом на более простой уровень перевода (перевод лишь отдельных словосочетаний, выдача подстроичника и др.). В [Машинный, 1987] предлагается (пока теоретически) проводить внутрисистемную оценку по основному параметру каждого уровня анализа (понимания). Этих параметров четыре: *непрерывность, грамматическая правильность, семантическая связность, осмысленность в данной ПО*. Они понимаются следующим образом. Непрерывность — это отсутствие вариантов перевода одной и той же единицы. Грамматическая правильность фразы — отсутствие нарушений законов построения синтаксической структуры переводящей фразы. Параметр связности применим к СемП целого текста; это отсутствие разрывов в семантическом графе (т. е. отсутствие нарушений правил семантической грамматики). Осмысленность в данной ПО — возможность непротиворечивых интерпретаций в единицах данной ПО структуры целого текста.

10.6. Перспективы развития систем машинного перевода

До конца 80-х годов ожидается наибольшее развитие систем типа АРМ. Они будут объединять в себе системы обработки текста (текстовые процессоры), доступ к удаленным ТБД и базам данных других типов, системы ЧМП, интерактивные (типа ALPS) и полностью автоматические СМП (см. § 10.2). Переводчик будет выбирать, какой режим работы ему больше всего подходит для перевода данного текста. Технология АРМ для переводчиков уже создана. Одна из новейших систем такого типа MELCOM-PSI (см. § 11.2).

Полностью автоматические СМП на больших ЭВМ будут работать в основном в крупных переводческих службах. Разрабатываются также такие версии

этих систем, которые станут доступными небольшим организациям и даже отдельным лицам. Специально для последних создаются «личные переводчики» — системы на персональных компьютерах для ведения деловой документации с одновременным переводом.

Что касается экономической эффективности (которая для всех СМП остается пока довольно низкой), то ожидается, что она будет повышаться по мере освоения технологии оптического ввода.

Однако новые информационные технологии существенно определили (и даже задержали) развитие лингвистической теории, особенно семантического уровня: наиболее сложные проблемы решает человек путем пред-, интер- и/или постредактирования или же он сам и переводит, пользуясь АРМ. Налицо большие достижения вычислительной лингвистики: морфологический анализ является уже неотъемлемой частью многих пакетов «текстовых процессоров», техника синтаксического анализа отточена и поддерживается сильными математическими вычислительными моделями. Но надежных пакетов «синтаксических анализаторов» еще нет, так как решение многих синтаксических проблем (типа обработки эллиптических предложений, сложных конструкций с однородностью и др.) определяется семантикой. Методика трансфера не дает решения и некоторых тривиальных, но частых переводческих соответствий (перевод информации об артикле, перевод грамматических времен и др.).

М. Нагао подчеркивает, что рано или поздно любая СМП достигает своего предела, за которым добиваться улучшений уже очень трудно; как скоро это произойдет, зависит от ее структуры. Поэтому в японские СМП с самого начала закладывается достаточно сложная и богатая лингвистическая теория (см., например, описание проекта Ми в [Нагао и др., 1986]). На первых порах такая теория не кажется эффективной по сравнению с простой языковой стратегией — ее преимущества начинают проявляться на сложном исходном материале.

У. Хатчинз [Hutchins, 1986] полагает, что в ближайшее время семантические проблемы вряд ли будут решены на таком уровне, что они могут серьезно улучшить МП.

Американские СМП придерживаются стандартной парадигмы АП — трехчастной системы с развитым трансфером, основывающейся на предположении, что между двумя языками существует легко обозримое структурное и достаточно простое лексическое соответствие. Такие системы моделируют работу мало-квалифицированного переводчика, осуществляющего пофразный перевод и мало заботящегося о сохранении единой связной линии текста; ее может восстановить постредактор: ведь и традиционный перевод нуждается в редактировании. Дж. Слокэм считает, что перевод через трансфер, осуществляющий поверхностно-синтаксические преобразования, может дать очень много: переводчики успешно справляются со своей задачей, не слишком вникая в смысл переводимого текста.

Конфигурация будущей СМП еще не найдена. Дальнейшее развитие МП сдерживается слабостью лингвистической теории. Сделанный в последние годы «прорыв» инженерии знаний в СМП пока мало что дает: не так просто включить в лингвистические синтаксические структуры «экстралингвистические знания», имеющие иную природу. Знания, поставляемые концептуальными структурами узких предметных областей, слишком ограничены и искусственны, чтобы можно было опереться на них при анализе реальных потоков текстов, относящихся, как правило, не к одной ПО. Объяснение семантики текста подчас существенно соотносится с ПО, но не сводится к ней. Нужна семантическая теория, которая выдерживает проверку на открытом множестве текстов, а не только в «лабораторных условиях».

Проблема перевода (и в частности, машинного перевода) — прежде всего лингвистическая проблема. Семантическая теория должна объяснять и законы развития связного текста, и соотношение текста с ПО, и коммуникативные свойства текста, и более поверхностные механизмы синтаксических и лексических замен.

В этом направлении развивается МП как научная область. Об этом свидетельствует возросший интерес к языкам-посредникам разного вида: концептуальным, в виде конкретного международного языка эсперанто, разного рода гибридным ЯП. Аппарат для лингвистически обоснованного ЯП предлагает, например, падежная грамматика Ч. Филлмора, являющаяся одной из самых популярных семантических моделей, но она объясняет только часть семантических процессов. Кроме того, нужна хорошая практическая теория лексикографии, теория текста, теория переводческого процесса в целом; нужны фундаментальные лингвистические работы. В этом отношении в отечественных разработках по МП достигнут высокий теоретический уровень, особенно в области синтаксиса (ЭТАП), лексикографии (словари систем АРАП, ЭТАП, ФРАП), логики естественного текста (см., например работу [Saluveer et al., 1986]).

Основной лингвистической проблемой остается формализованное представление семантики текста, включение концептуальных знаний в его семантические структуры. Важно, что проблемы поставлены, а многие трудности и узкие места диагностированы. Их решение, по-видимому, будет найдено в рамках общего развития искусственного интеллекта, где все глубже и шире осознается важность лингвистических аспектов исследования.

У. Хатчинз называет среди главных проблем: уточнение многоуровневых представлений с точки зрения их содержания и структуры; способ интеграции разных уровней «понимания», развитие простых и гибких методов анализа, справляющихся с неполнотой, метафорическим употреблением и семантическими нарушениями; развитие формализмов для текстовой семантики, обработка больших и сложных семантических сетей, обеспечение хороших принципов кодирования, особенно для больших словарей.

Дж. Слокэм подчеркивает еще один важный аспект развития больших систем — необходимость автоматизировать сам процесс исследований. Он считает, что существующие (ручные) методы построения семантико-когнитивных моделей явно неприемлемы для использования в крупных системах [Slocum, 1985].

Действительно, решение многих проблем МП связано с созданием благоприятных условий для развития самого лингвистического обеспечения — грамматик, анализаторов и синтезаторов текста, а также словарей. Именно с этой целью за рубежом уделяется большое внимание созданию автоматизированных рабочих мест (Workstations) для лингвистов. Примерами такой программной среды, обеспечивающей лингвистические исследования, являются АРМ SAFRAN (Саарбрюкен), АРМ системы METAL, программная среда системы GETA (Гренобль).

При всей важности лингвистического фундамента работа современных крупномасштабных СМП не может основываться только на лингвистической теории или на одной единственной лингвистической теории. Современные СМП — это комплексные системы, которые могут опираться на разные лингвистические теории и иметь выход к разным информационным задачам.

Глава 11.

Современное состояние машинного перевода

Н. Н. Леонтьева, З. М. Шалапина

11.1. Общие сведения о современных системах машинного перевода

Известно несколько десятков систем МП, находящихся на разных стадиях разработки или эксплуатации. Точное их число назвать трудно: во-первых, имеющиеся данные, по-видимому, неполны; во-вторых, когда в литературе описываются многоязычные средства МП, разрабатываемые одной и той же фирмой

на базе одного и того же типа вычислительной техники, их соотношение между собой нередко не уточняется. Поэтому трудно установить, идет ли речь о совокупности нескольких более или менее самостоятельных систем (отличающихся друг от друга достаточно существенными деталями своего математического и/или лингвистического обеспечения) или о разных версиях одной системы (где, несмотря на различие описываемых языков, все сколько-нибудь принципиальные вопросы описания этих языков и обработки текстов решаются вполне единообразно). Ниже при рассмотрении таких систем используется следующий принцип: если система выделена среди других разработок тех же авторов особым названием, она описывается отдельно; в остальных случаях дается сводное описание всех систем (версий системы), объединенных общим названием.

Учитываемые здесь системы МП (всего более 70) и их основные характеристики представлены в табл. 11.1. В основном в нее включены системы, функционирующие или разрабатываемые в настоящее время. Из систем, которые уже прекратили свое существование, в таблице указаны только те, которые благодаря проведенным в их рамках теоретическим и экспериментальным исследованиям имели существенное значение для последующих разработок. Ныне существующие системы разбиты в данной главе на три группы: промышленные, развивающиеся и экспериментальные, ниже каждая из них будет описана отдельно.

11.2. Промышленные системы машинного перевода

К промышленным отнесены те системы МП, которые либо имеются в продаже (ALPS, ATLAS, LOGOS, MELCOM-PSI, METAL, Micro-Pak, PIVOT, SMART, SYSTRAN), либо постоянно и в относительно широких масштабах эксплуатируются в настоящее время в одной или нескольких организациях (CULT, ENGSPAN, SPANAM, TAUM-METEO, TAURAS, TITRAN, TITUS, КНР-1, КНР-2, а также отечественные АНРАП, СИЛОД, ЧПИ). Все они ориентированы прежде всего на техническую и другую специализированную документацию. Основные тематические области: электроника, электротехника, вычислительная техника, информатика, технология программирования, авиатехника, сельское хозяйство, химия, металлургия, автомобильная и пищевая промышленность, транспорт, военная, экономическая, финансовая документация, административные материалы, материалы по медицине и здравоохранению, по охране окружающей среды и т. п. Ряд систем (в частности, ATLAS-II и СИЛОД) имеют также словари общественно-политической и публицистической лексики.

Методологические основы перечисленных систем весьма разнообразны: в одних (например, в ATLAS-I, SPANAM, TAUM-METEO, АНРАП, во многих системах серии Micro-Pak Yoko-Yoko и SYSTRAN) используется прямой подход к процессу перевода, в других (как в ATLAS-II и METAL) — стандартная «трехэтапная» стратегия с разделением анализа, межъязыковых операций (трансфера) и синтеза, в третьих (PIVOT), по сообщениям их авторов, — концепция языка-посредника. Имеются в числе промышленных систем и системы человеко-машинного типа (ALPS, CULT, TITUS, КНР-2, ЧПИ), системы, по-разному совмещающие отдельные аспекты прямого и «трехэтапного» подходов (TAURAS, TITRAN), и др. Поэтому не будем выделять в данной группе систем какие-либо более мелкие их классы (такая классификация при отмеченном разнообразии реализуемых ими подходов носила бы, скорее всего, более или менее искусственный характер), а рассмотрим все эти системы в алфавитном порядке следования их названий и дадим для каждой из них краткую характеристику тех основных принципов, на которых она строится, и тех возможностей, которые она предоставляет своим пользователям¹.

¹ Сведения об описываемых системах, которые включены в табл. 11.1, не повторяются в тексте.

Основные системы МП и их характеристики

Название системы ¹ и раздел, где она описана ²	Страна, организация-раз- работчик	Языки и направление перевода	Современное состояние и период разработки	Основные источники
ALPS (§ 11.2)	США, Ун-т Бриггема Янга, затем фирма ALPS	С английского на фран- цузский, немецкий, ис- панский, португальский; ис- есть сообщения о под- ключении арабского и других языков	Имеется в продаже с 1980 г.; явилась дальней- шим развитием разрабо- ток по системе ITS (см. ниже)	Slocum, 1985; Hutchins, 1986
ATHENE/E, коммерче- ское название HICATS/EJ (§ 11.3)	Япония, фирма Хитати	С английского на япон- ский	Начало разработок — 1978 г. ³	Кадзи и др., 1985
ATHENE/N (§ 11.3)	Япония, фирма Хитати	С японского на англий- ский	Начало разработок — 1978 г.	Кадзи и др., 1985
ATLAS-I (§ 11.2)	Япония, фирма Фудзину	С английского на япон- ский	Имеется в продаже с сентября 1984 г.	Kushima et al., 1986
ATLAS-II (§ 11.2)	То же	С японского на англий- ский	Имеется в продаже с июня 1985 г.	Утида, 1985; McWilliams, 1986a McWilliams, 1986b
CETA	Франция, Гренобльский ун-т	С японского на француз- ский и немецкий, с ан- глийского на японский, французский, немецкий	Имеются модельные си- стемы	Slocum, 1985; Vauquois et al., 1985
		С русского на француз- ский	Разрабатывалась в 1961—1971 гг.; опыт ис- пользован в разработке системы GETA (см. ни- же)	

CULT (\$ 11.2)	Гонконг, Китайский ун-т, финансирующие организации — Азиатский фонд и фонд Рокфеллеров	С китайского на английский С английского на китайский	Начало разработок — 1968 г., начало промышленной эксплуатации — 1975 г. Находится в процессе разработки	Slocum, 1985. Hutchins, 1986
DLT (\$ 11.4)	Нидерланды, Бюро разработки систем (Утрехт); финансирует Комиссия европейских сообществ	Многоязычная система, рассчитанная на английский, немецкий, французский и итальянский языки; в перспективе также на японский, китайский, арабский	Начало исследований — 1979 г., начало практических разработок — 1985 г.; планируется ввести в промышленную эксплуатацию к 1990 г.	Papagaaij, 1986
ENGSPAN (\$ 11.2)	США, Панамериканская организация здравоохранения (Вашингтон)	С английского на испанский	Начало разработок — 1981 г., начало промышленной эксплуатации — 1985 г.; учтен опыт системы SPANAM (см. ниже)	Slocum, 1985
EUROTRA (\$ 11.3)	ЕЭС	7 рабочих языков ЕЭС в обоих направлениях для каждой пары языков	Начало исследований — 1978 г., начало практических разработок — 1982 г.; планируются сроки создания и опробования первого прототипа — 1987—1988 гг.	Johnson et al., 1985; Ариолд, 1986
GAT	США, Джорджаунский ун-т	С русского на английский	Начало разработок — 1954 г.; период промышленной эксплуатации — 1964—1979 гг.	Slocum, 1985; Hutchins, 1986; Кулагина, 1979

Название системы ¹ и раздел, где она описана ²	Страна, организация-разработчик	Языки и направление перевода	Современное состояние и период разработки	Основные источники
GETA, весь алгоритмический комплекс ARIANE-78, основная версия RUS-CRA (\$ 11.3)	Франция, Гренобльский ун-т	По замыслу многоязычная система; основная версия — с русского на французский	Начало разработок — 1971 г.; учтен опыт работы по системе CETA (см. выше); в современном виде разрабатывается с 1978 г.	Vauquois et al., 1985
НIT (\$ 11.4)	КНР, Харбинский Технологический ин-т	С русского на китайский	Начало разработок — 1979 г.; экспериментальная версия подготовлена к 1981 г.	Ван Чжень, доклад в МГУ, 1987
ITS	США, ун-т Бригема Янга	С английского на французский, испанский, португальский, немецкий, китайский	Начало разработок — 1970 г.; окончательным результатом явилось создание серии систем ALPS (см. выше)	Melby, 1982; Slocum, 1985; Hutchins, 1986
KBMT (\$ 11.4)	США, ун-т Карнеги — Меллона (Питтсбург)	По замыслу многоязычная система; основная версия — с английского на японский	Реализована в экспериментальном варианте	Tomita et al., 1986
LOGOS (\$ 11.2)	США, фирма LOGOS DEVELOPMENT; финансирующие организации: до 1974 г. BBC США, 1978—1981 гг. — фирма Siemens AG, позже фирма Wang	С английского на венесуэльский, французский, испанский, немецкий, персидский С немецкого на английский	Начало разработок — 1964 г.; период эксплуатации — 1971—1973 гг. Экспериментальные модели, разрабатывавшиеся в середине 70-х годов	Hutchins, 1986
			Начало разработки — 1978 г.; имеется в продаже	

LUTE (\$ 11.4)	Япония, фирма NTT	С японского на английский и с английского на японский	Реализована в экспериментальном варианте	Numura et al., 1986
MELCOM-PSI	Япония, фирма Мицубиси	С японского на английский	Имеется в продаже с января 1987 г.	Мицубиси, 1987
METAL, весь переводческий комплекс — LRC, с 1980 г. финансирующая организация — фирма Siemens AG	США, Техасский ун-т, с 1980 г. финансирующая организация — фирма Siemens AG	С немецкого на английский	Начало исследований — 1961 г., начало конкретных разработок — 1975 г.; имеется в продаже с апреля 1985 г.	Slocum et al., 1984
		С английского на немецкий и испанский; с немецкого на китайский и испанский	Находятся в процессе разработки	
Micro-Pak (\$ 11.2)	Япония — США, фирма Bravice International & Weidner Communications	С японского на английский	Имеется в продаже с октября 1985 г., демонстрировалась на выставке ЯПОНИЯ-86 (Москва, октябрь 1986 г.)	Bravice, 1986
		С японского на корейский	Поступила в продажу в 1986 г.	
		С английского на японский, китайский, корейский	Должны поступить в продажу в 1987—1988 гг.	
Micro-Pak серия Yoko-Yoko		С английского на немецкий, французский, испанский, португальский, итальянский, арабский; с французского, немецкого, испанского на английский	Имеются в продаже с октября 1985 г.; испанско-английская и англо-французская системы деструктурировались на выставке Япония-86 (Москва, октябрь 1986 г.)	
MOPTRANS (\$ 11.4)	США, Йельский ун-т (Нью-Хейвен)	С испанского на английский	Ведутся теоретические и экспериментальные исследования	Hutchins, 1986

Название системы ¹ и раздел, где она описана ²	Страна, организация-раз- работчик	Языки и направление перевода	Современное состояние и период разработки	Основные источники
Ми (\$ 11.3)	Япония, Киотоский ун-т, Электротехнические ла- боратории, Японский ин- формационный центр по науке и технике. Систе- ма исследований по об- работке информации; ра- бота ведется в рамках японского правительст- венного проекта по МП	С японского на англи- ский и с английского на японский	Начало исследований — 1964 г., начало работ конкретно по проекту Ми — 1982 г.; к 1986 г. создана система-про- тип; планируемый срок создания промышленной версии — 1990 г.	Нагао и др., 6
NARA (\$ 11.3)	Япония, Токийский ун-т	С корейского на япон- ский и с японского на корейский	Разработана система- прототип	Chung et al., 1986
NTT-CSK* (\$ 11.4)	Япония, фирма NTT сов- местно с исследовател- ским ин-том CSK	С английского на япон- ский и с японского на английский	Находится в процессе разработки	Kudo et al., 1986
PERSIS (\$ 11.4)	Япония, ун-т Кобе	С персидского на англи- ский	К августу 1986 г. прове- дены первые эксперименты	Samatrad et al., 1986
PIVOT, эксперименталь- ный комплекс VENUS (\$ 11.2)	Япония, фирма NEC	С японского на англи- ский и с английского на японский	Имеется в продаже с 1986 г.	Мураки, 1985; Harada, 1986; McWilliams, 1986b
Rosetta (\$ 11.4)	Нидерланды, фирма Philips	По замыслу многоязыч- ная система, рассчитан- ная на голландский, ан- глийский, испанский и другие языки	Начало исследований — 1980 г.	Appelo, 1986

SALAT (§ 11.4)	ФРГ, Ин-т прикладной лингвистики Гейдельбергского ун-та	По замыслу многоязычная система, рассчитанная на немецкий, французский, английский и другие языки	Начало исследований — 1978 г.	Hutchins, 1986
SCANLAN (§ 11.3)	ПНР, Политехнический ин-т, гор. Жешова	С русского на польский	Планируемый срок ввода в промышленную эксплуатацию — конец 1989 г.	Hirpe et al., 1987
SEMSYN (§ 11.4)	Япония — ФРГ, фирма Фудзиси и Штутгартский ун-т; финансирует Мин-во исследований и технологии ФРГ	С японского на немецкий	Начало разработок — 1983 г.; к 1986 г. создана система-прототип	Rosner, 1986
SHALT (§ 11.3)	Япония, фирма IBM — Japan	С английского на японский	Начало разработок — 1982 г.; к 1986 г. создана система-прототип	Tsutsumi, 1986
SMART (§ 11.2)	США, фирма Smart AI inc.	С английского на французский	Имеется в продаже с 1978 г.; начало промышленной эксплуатации — 1983 г.	Hutchins, 1986
SMU (§ 11.3)	Малайзия, ун-т Сен-Малайзия (Пенанг)	С английского на малайский	Начало разработок (на основе комплекса ARIANE-78, см. выше GETA) — 1978 г.; в 1980 г. создана система-прототип	Tong, 1986
SPANAM, по названию организации-разработчика — РАНО (§ 11.2)	США, Панамериканская организация здравоохранения (Вашингтон)	С испанского на английский	Начало разработок — 1976 г.; начало промышленной эксплуатации — 1980 г.	Slocum, 1985; Tucker, 1984

Название системы ¹ и раздел, где она описана ²	Страна, организация-раз- работчик	Языки и направление перевода	Современное состояние и период разработки	Основные источники
SUSY (§ 11.4)	ФРГ, Саарский ун-т (Са- арбрюкен)	С русского и английско- го на немецкий; прово- дились эксперименты также с эсперанто, дат- ским, французским язы- ками	Начало исследований — 60-е годы; начало работ над 1-й версией — 70-е годы, над 2-й — 1981 г.	Luckhart et al., 1983
SYSTRAN (§ 11.2)	США, фирма LATSEC inc.	С русского на англий- ский, с английского на французский, немецкий, русский, итальянский, португальский, японский, арабский; с немецкого на английский, французский, испанский; с француз- ского на английский	Начало разработок — 1964 г.; начало промыш- ленной эксплуатации — 1970 г.	Кулагина, 1979; Slocum, 1985; Hutchins, 1986
TAS-1 (§ 11.3)	ГДР, Ун-т Карла Марк- са (Лейпциг)	С английского на немец- кий	Находится в процессе разработки	Shreve et al., 1986
TAUM-AVIATION	Канада, Монреальский ун-т	С английского на фран- цузский	Начало разработок — 1977 г.; в 1981 г. работы прекращены	Isabelle et al., 1985
TAUM-METEO (§ 11.2)	То же	То же	Начало разработок — 1975 г.; начало промыш- ленной эксплуатации — 1976 г.	То же
TAURAS, коммерческое название AS-TRANSAC (§ 11.2)	Япония, фирма Тосйба	С английского на япон- ский	Имеется в продаже с 1986 г.	Атапо, 1986, McWilliams, 1986

THALIA-3, коммерческое название MELTRAN (§ 11.3)	Япония, Ин-т информации и электроники	С японского на английский	Имеется система-прототип; планируемый срок ввода в промышленную эксплуатацию — 1988 г.	Окаси и др., 1986
TITRAN (§ 11.2)	Япония, Киотский ун-т	С японского на английский	К 1983 г. разработана промышленная система-прототип	Ibuki et al., 1983
TIT-SUT* (§ 11.4)	Япония, Токийский ин-т технологии и Токийский научный ун-т	С английского на японский	Имеется экспериментальная модель	Танака и др., 1985
TITUS (§ 11.2)	Франция, Текстильный ин-т Франции	Многоязычная система, работающая с английским, французским, немецким, испанским языками	Начало эксплуатации — 1970 г.; последняя версия реализована в 1980 г.	Hutchins, 1986
TRANSLATOR (§ 11.4)	США, Колгейтский ун-т	По замыслу многоязычная система, рассчитанная на английский, японский, испанский, русский и другие языки	Начало разработок — 1983 г.	Nirenburg et al., 1986
TRANSOFT (§ 11.3)	США — ФРГ, Медицинские учреждения Джона Хопкинса (Балтимор), Массачусетская клиника больница (Бостон), Фрайбургский ун-т (Фрайбург)	С немецкого, французского, итальянского, испанского, турецкого на английский	Находится в процессе разработки	Moore et al., 1986
UMIST (§ 11.4)	Великобритания, Манчестерский ун-т	По замыслу многоязычная система; основная версия — с английского на японский	Имеется система-прототип	Whitelock et al., 1986

Название системы ¹ и раздел, где она описана ²	Страна, организация-раз- работчик	Языки и направление перевода	Современное состояние и период разработки	Основные источники
UOP * (\$ 11.4)	Япония, ун-т префекту- ры Осака	По замыслу многоязыч- ная система; основная версия — с английского на японский	Имеется эксперименталь- ная система	Nishida et al., 1986
WEIDNER	США, фирма Weidner Communications	С английского на испан- ский, французский, не- мецкий; итальянский, португальский, арабский, японский; с испанского на английский; с фран- цузского на английский и испанский; с немецкого на английский и фран- цузский; с японского на английский	Начало разработок — 1977 г.; начало промыш- ленной эксплуатации — 1980 г.; с 1983 г. фирма вошла в состав объеди- ненной фирмы Bravice International & Weidner Communications (см. вы- ше Micro-Pak)	Sissum, 1985; Hutchins, 1986
АМПАР (\$ 12.2)	СССР, ВЦП ГКНТ и АН СССР (Москва)	С английского на рус- ский	Начало разработок — ко- нец 50-х годов; начало промышленной эксплуа- тации — 1981 г.	Марчук, 1983; Тихомиров, 1984
АНАП (\$ 12.2)	То же	С английского и немец- кого на русский	Начало разработок — 1986 г.	Тихомиров, 1987
АПАЧ (\$ 11.3)	ЧССР, Карлов ун-т (Прага)	С английского на чеш- ский	Начало разработки — 1976 г.	Kirschner, 1982
АРАП	СССР, МГПИИЯ им. М. Мореза (Москва)	С английского на рус- ский	Начало исследований — 1970 г.; в 1977 работы прекращены в связи с отсутствием технической базы	Шалапина, 1974; Шалапина, 1977
ИПКИР * (\$ 11.4)	СССР, ИПКИР ГКНТ и АН СССР (Москва)	С английского на рус- ский	Ведутся теоретические исследования	Певнер, 1981; ВЦП, 1979, с. 30—31

КНР-1* (§ 11.2)	КНР, Пекинский ун-т, Институт научно-технической информации (Пекин)	С английского на китайский С японского на китайский	Начало промышленной эксплуатации — 1984 г. Планируется разработать к 1990 г.	Ван Чжень, 1987
КНР-2* (§ 11.2)	КНР, НИИ «Искра» (Пекин)	С английского на китайский	Начало промышленной эксплуатации — 1985 г.	То же
ЛГУ*	СССР, Ленинградский гос. ун-т	С английского на русский	Начало разработок — 1961 г.; первое экспериментальное опробование системы — 1967 г.	Лейкина и др., 1966
НЕРПА (§ 12.2)	СССР, ВЦП ГКНТ и АН СССР (Москва)	С немецкого на русский	Начало разработок — 1976 г.; сдача в промышленную эксплуатацию — 1985 г.	Тахомиров, 1987
ПАРС (§ 11.3)	СССР, ВНИИТЭлектромаш (Харьков), Харьковский гос. ун-т	С английского на русский	Имеется экспериментальная версия системы, узланная с системой автоматического индексирования	Войнов и др., 1987
ПЕРЕВОДЧИК (§ 11.3)	СССР, МПГИИЯ (Минск)	С русского на английский и французский; с английского и французского на русский	Имеется экспериментальная версия системы	Международный 1983, с. 43—44
РУСЛАН (§ 11.3)	ЧССР, Карлов ун-т (Прага)	С чешского на русский	Начало разработок — 1985 г.; планируется ввести в промышленную эксплуатацию в 1990 г.	Hajšová et al., 1981
СИЛОД (§ 11.2)	СССР, ЛГПИ им. Герцена (Ленинград)	Многоязычная система, рассчитанная на английский, испанский, французский, китайский, турецкий, японский и русский	Находится в промышленной эксплуатации	Международный, 1983, с. 37—40; Международная. 1985, с. 130—131

Название системы ¹ и раздел, где она описана ²	Страна, организация-разработчик	Язык и направление перевода	Современное состояние и период разработки	Основные источники
СИМПАР (§ 11.3)	СССР, ВНИИПНефтехим (Киев)	С английского на русский	Начало экспериментальной эксплуатации — 1981 г.	Мирам и др., 1982
ФРАП (§ 12.4)	СССР, ВЦП ГКНТ и АН СССР (Москва)	С французского на русский	Начало разработок — 1976 г.; начало экспериментальной эксплуатации — 1980 г.	Леонтьева и др., 1980; Синтаксический, 1981; Машиный, 1987
ФР-II (§ 11.4)	СССР, Ин-т прикладной математики АН СССР	С французского на русский	Начало экспериментальных исследований — 1971 г.	Кулагина, 1979
ЧПИ* (§ 11.2)	СССР, Чимкентский педагогический ин-т, Казахский химико-технологический ин-т (Чимкент)	С английского на русский	Начало опытно-промышленной эксплуатации — 1977 г.	Международный, 1983. с. 36—39; Международная, 1985. с. 18—19
ЭТАП-1 (§ 12.3)	СССР, ИНФОРМЭЛЕКТРО (Москва)	С французского на русский	Начало разработок — 1974 г.; начало экспериментальной эксплуатации — 1980 г.	Апресян и др., 1983
ЭТАП-2 (§ 11.3)	То же	С английского на русский	Начало разработок — 1981 г.; начало опытно-промышленной эксплуатации — 1985 г.	Апресян и др., 1988
ЯРАП (§ 11.3)	СССР, Ин-т востоковедения АН СССР, ВЦП ГКНТ и АН СССР	С японского на русский	Начало исследований в ИВ АН СССР — 1978 г.; начало совместных работ по реализации — 1986 г.	Шаляпина, 1980

¹ Для систем, коммерческая и прототипическая или экспериментальная версия которых имеют разные названия, указываются (по возможности, с пояснениями) все названия.

² Отсылка к номеру раздела дается только для систем, получающих в соответствующем разделе отдельное описание; иначе она указывает на тип системы — принадлежность ее к числу промышленных (§ 11.2, § 12.2), развивающихся (§ 11.3, § 12.4) или экспериментальных (§ 11.4, § 12.3). Отсутствие такой отсылки означает, что система уже прекратила свое существование.

³ Отсутствие сведений о годе окончания разработки системы означает здесь и далее, что работа над этой системой продолжается.

Системы серии ALPS (Automated Language Processing Systems, США). В основу серии легли результаты работ над системой ITS (Interactive Translation System) в университете Бригема Янга. Затраты на ее разработку к 1987 г. составляли 350 чел.-лет (\$ 16 млн.). Лингвистический аппарат базируется на «грамматике юнкции» [Lytle, 1980] — одной из модификаций ТФГ. Программы написаны на специализированном языке программирования высокого уровня Eclipse. Системы могут функционировать в трех режимах ЧМП: экранного редактирования и словарного поиска по запросу переводчика; «пословного перевода» — прямой подстановки вместо слов, идиом и терминологических сочетаний, имеющихся в оригинале, их переводных эквивалентов; автоматизированного пофразного перевода с участием человека. Последний режим фактически почти не применяется. Основные пользователи: бюро переводов самой фирмы ALPS, компании Xerox, Computer Vision, Control Data (во Франции), IBM (в Италии) Hewlett-Packard (в Мексике).

Система ATLAS-I (Япония). Перевод осуществляется в рамках прямого синтаксического подхода с использованием «встроенных» словарей. Грамматика непосредственно включена в программное обеспечение. Снятие лексической омонимии и определение синтаксических функций слов входного текста совмещено с процедурами выбора их переводных эквивалентов, и вся предусматриваемая для этого семантическая информация приписана только самим имеющимся эквивалентам, но не единицам входного языка. Синтез сводится к вставке служебных элементов и к отдельным позиционным преобразованиям; в структурно-синтаксическом отношении выходные фразы в основном калькируют конструкции оригинала. Имеются средства постредактирования, позволяющие выводить на дисплей одновременно входное и выходное предложения, а также использованные при переводе словарные описания, так что редактирование перевода может быть увязано с коррекцией словарей. Система реализована на базе больших ЭВМ серии FACOM M (типа M340R и выше) и мини-ЭВМ серии FACOM S3000. Скорость работы — 60 000 слов в час, стоимость аренды — 350 000 иен в месяц, стоимость дополнительного словаря — 150 000 иен (данные 1984 г.). Используется более чем 70 организациями.

Система ATLAS-II (Япония). Перевод осуществляется через этап межязыковых операций. В ходе анализа для входного предложения строится его «концептуальная структура» — семантическая сеть, содержащая сведения о морфологических характеристиках имеющихся в нем слов, о падежных отношениях между ними и о некоторых аспектах организации их значений. Поверхностно-синтаксическая информация в такой структуре в основном отсутствует, чтоближает ее с понятием языка-посредника и упрощает структурные преобразования при переводе. В результате на этой основе можно получать перевод входного текста на разные языки (см., например, описание системы SEMSYN в § 11.4) и строить более «естественные» выходные фразы; в то же время возрастает неточность перевода. Помимо собственно лингвистической информации (прежде всего сведений о сочетаемости слов в предложении), в системе используется семантическая информация, извлекаемая из «модели мира» (сведения о причинно-следственных отношениях, требования «здравого смысла» и т. п.). Словари входного (входных) и выходного (выходных) языков отделены друг от друга, а в пределах каждого из них общепотребительная лексика отделена от терминологической. Словари общепотребительной лексики японского и английского языков содержат примерно по 60 000 единиц.

Система реализована на базе больших ЭВМ фирмы Фудзиси серии FACOM M (OS IV/F4MSP) и обеспечивает скорость перевода 60 000 слов в час (на ЭВМ FACOM M380). На выставке EXPO'85 (Цукуба, Япония) с ее помощью в течение шести месяцев осуществлялся непрерывный перевод с английского на японский, французский и немецкий языки сообщений агентства «Франс-пресс» (с выводом результатов на дисплей). Кроме того, посетители могли предлагать японские или английские фразы, которые после предредактирования переводились соответственно на английский или японский, а также на французский и немецкий языки. Система приобретена более чем 30 организациями.

Система CULT (Chinese University Language Translator, Гонконг). Это система ЧМП, предусматривающая существенное предредактирование входных текстов (вплоть до проставления в них границ придаточных и отдельных составляющих) и определенный объем интер- и постредактирования (за счет которого обеспечивается снятие лексической и синтаксической неоднозначности, расстановка артиклей, уточнение грамматического числа, уточнение залога и наклонения глаголов и т. п.). С 1978 г. все эти процедуры, как и ввод входных китайских текстов, осуществляются с помощью средств экранного редактирования. Собственно переводческий компонент системы базируется на стратегии прямого перевода. Анализ ограничен словарным поиском, идентификацией именных и глагольных групп, характерных для китайского языка модификаторов и служебных слов, выявлением некоторых грамматических категорий и (частично) определением отношений между модификаторами и модифицируемыми ими группами. Длинные фразы переводятся по частям. Система реализована на ЭВМ ICL 1904A и используется для регулярного перевода на английский язык пекинских журналов «Acta Mathematica Sinica» и «Acta Physica Sinica». Продолжаются работы по расширению и совершенствованию словарей системы. Разрабатывается также ее англо-китайская версия.

Система ENGSPAN (ENGLISH-SPANISH, США). Разработана с учетом опыта создания и эксплуатации системы SPANAM (см. ниже), но использует более современную технологию (в частности, РСП-грамматики). Применяется четырьмя переводчиками Панамериканской организации здравоохранения.

Системы серии LOGOS (США). Ориентированы на узкоспециальные технические области. О лингвистическом аппарате известно только, что его общая концепция близка к принципам падежной грамматики и теории валентностей. Для обрабатываемых предложений строится древесное представление семантико-синтаксической структуры, что позволяет предположить наличие этапа межязыковых операций, выполняемых на уровне таких структур. Немецко-английская система данной серии реализована на мини-ЭВМ фирмы Wang; по оценкам 1982—1983 гг. она дает удовлетворительный перевод для 60—80 % всего входного текста. Система приобретена и используется, в частности, корпорациями Nixdorf, Triumph-Adler, Hewlett-Packard. Продолжается работа над англо-французской, англо-немецкой и другими системами.

Комплекс MELCOM-PSI (Япония). Представляет собой автоматизированное рабочее место, в комплект поставки которого входит система англо-японского АП. Словари системы содержат 50 000 общеупотребительных слов и 50 000 терминов по обработке информации. Большая часть словарей хранится в основной памяти. Предусмотрен семантический анализ переводимого текста (видимо, пофразный). Язык программирования — ESP (Extended Self-contained PROLOG). Скорость перевода — 50 000 слов в час. Имеются развитые средства пред- и постредактирования и форматирования текста.

Система METAL (MEchanical Translation and Analysis of Languages), или LITRAS (США). Имеет многоязычную ориентацию. Перевод осуществляется через этап межязыковых операций, полностью отделенный от этапов анализа и синтеза. Программное обеспечение (написанное на Лиспе) допускает описание разных языков в рамках разных лингвистических концепций. Так, для немецкого языка подсистема анализа (около 500 правил) построена на основе КС-грамматики, расширенной произвольными трансформациями. Для английского языка при анализе используется модификация обобщенной НСГ без трансформаций. Применение каждого правила сопровождается вызовом «встроенной» в него процедуры, обеспечивающей проверку релевантных для него синтаксических и семантических ограничений, интерпретацию обрабатываемой составляющей (с учетом падежных зависимостей), возможно, изменение ее структуры, а также оценку степени ее приоритетности по сравнению с имеющимися альтернативами. Подсистема межязыковых операций включает два пакета трансформаций: для грамматики и для словаря. Правила обоих типов выполняются за один проход дерева, полученного при анализе, в направлении сверху вниз. Конкретный порядок обхода дерева задается самими правилами. Выбор этих правил обычно

контролируется отсылками, приписанными правилам, работающим при анализе, но для перевода некоторых конструкций (например, немецких придаточных) могут строиться и грамматике более общего типа, не использующие таких отсылок. В словарях системы самостоятельные статьи имеют, кроме лексических основ, также аффиксы и знаки препинания. При проверке работы системы на материале 50 страниц немецкого текста было правильно переведено 83 % предложений.

Системы серии Micro-Pak (Япония — США). В системах типа Yoko-Yoko данной серии (изначально разрабатывавшихся корпорацией Weidner) применяется прямой подход к переводу с локальной обработкой текста на уровне именных и глагольных словосочетаний и строгим разделением четырех ее этапов: снятия омографии, выделения идиоматических оборотов, структурного анализа и перевода к выходному языку. Решения, принятые на том или ином этапе, могут пересматриваться по данным, получаемым позднее. В японско-английской системе предусмотрен *глобальный анализ* структуры фразы.

В разрабатываемой англо-японской системе, по данным 1986 г., планируется также учет сведений о соседних предложениях текста. Для случаев расхождения двух языков в отношении типичных для них способов описания идентичных ситуаций в нее вводятся и определенные элементы анализа понятий (на базе соответствующей словарной информации). В англо-китайской системе анализ должен опираться на аппарат ЛФГ; при наличии альтернатив должны применяться весовые оценки.

Существующие системы реализованы на мини-ЭВМ корпорации DEC серии VAX (язык программирования Фортран-77 в ОС VMX) и на ЭВМ IBM PC/XT, AT, 5550 (язык C в ОС MS-DOS). В 1986 г. сообщалось о подготовке к переходу на основную модель ЭВМ фирмы IBM, обещающему ускорение процесса АП в 8—10 раз. Максимальная скорость перевода по состоянию на 1986 г. — 8000 слов выходного текста в час; ее имеет англо-французская система при работе на мини-ЭВМ VAX 865 с одним пользователем (требуемая емкость оперативной памяти — 380 Кбайт, дисковой — 10 Мбайт). Минимальная скорость — 1000 слов выходного текста в час — достигается японско-английской системой на IBM 5550; она требует не менее 640 Кбайт оперативной и 20 Мбайт дисковой памяти.

Наибольший спрос имеет японско-английская система (по состоянию на конец 1985 г. ее приобрели более 900 пользователей; на все прочие системы нашлось лишь около 100 покупателей). Словари системы содержат 60 000 единиц, среди них около 7000 словосочетаний. Пользователь может добавлять до 40 000 новых словарных единиц, а также менять или устранять имеющиеся, исключая 8000 единиц, для которых заданы специфические процедуры обработки. Для входных текстов требуется некоторое предредактирование. Вводятся они методом «кана-кандзи» или «ромадзи-кандзи»: оператор набирает транскрипцию текста в японской слоговой азбуке (кана) или латинице (ромадзи), и на экране автоматически формируется иероглифическая запись (кандзи). При ее неоднозначности даются все допустимые варианты. Для операторов-японцев, хорошо владеющих родной письменностью, этот метод ввода, по их оценкам, более чем в 1,5 раза эффективнее других; для неаппонцев он практически непригоден. В ходе АП слова, не найденные в словаре, могут выводиться на дисплей с тем, чтобы пользователь приписал им нужную информацию; далее она может автоматически переноситься в словарь системы. Выходной текст выдается для редактирования в нижней части дисплея. Разработана специальная методика обучения пользователей работе с данной и другими системами серии, реализованная на 16-битовом микрокомпьютере.

Система PIVOT (в экспериментальном варианте VENUS — VEHICLE for Natural language Understanding and Synthesis, Япония). Перевод осуществляется через так называемую понятийную структуру, организованную как инстанция базы знаний и задающую только «коммуникативное содержание» предложения без сведений о его синтаксисе и лексике; разработчики расценивают ее как языкопосредник (pivot).

В процессе анализа выделяются три этапа: «морфологический» (сегментация предложения на слова, определение их частей речи и установление межсловных зависимостей), синтактико-семантический (построение «семантического» дерева зависимостей, содержащего все слова предложения; используются расширенная грамматика зависимостей и экспертные сведения о предметной области, имеющиеся в базе данных) и понятийный (преобразование полученного дерева в сетевую понятийную структуру, элементами которой являются уже не слова, а «концепты»). Аналогичные три этапа синтеза — это переход от «концептов» к лексике выходного языка, построение древесной (синтактико-) «семантической» структуры выходного предложения и формирование по ней окончательной цепочки словоформ. И при анализе, и при синтезе работают одни и те же словари, разбитые для каждого из двух языков на три подмассива (связанных между собой указателями): морфологический, синтаксический и семантический.

Система реализована на базе ОС общего назначения корпорации NEC; ACOS-6/MVX и ACOS-4/MVP XE; может работать в пакетном или интерактивном режиме. Для ввода, пред-, интер- и постредактирования текстов, а также для составления и редактирования словарей имеются диалоговые средства, работающие под управлением меню. Система может подключаться к различным конторским системам автоматизированной обработки документации. Сообщается о возможности ввода текстов с голоса. В качестве дальнейшей перспективы на выставке EXPO'85 в Цукубе была показана небольшая система устного АП для диалогов на тему «Как проехать по Токио»; система была чисто демонстрационной, рассчитанной на перевод не более 500 различных фраз на базе словаря из 120 слов.

Пример перевода, обеспечиваемого системой PIVOT: Кикай хонъяку сисутэ-му но кайхацу ва саякй но кансин то китай ни сасазара, кю: питти-ни сусундэ иру («Разработка систем АП стимулируется интересом и надеждами общества и продвигается быстрыми темпами») ⇒ Interest and expectation of society supports and development of the machine translation system proceeds rapidly.

Система SMART (США) — политематическая система, основанная на технологии ИИ. В [Hutchins, 1986] она характеризуется как «экспертная система для перевода». О лингвистическом обеспечении известно только, что синтаксис описывается в терминах последовательностей именных синтагм с опорой на развитые стандартизированные словари, насчитывающие свыше 10 000 входов. Система реализована на ЭВМ серии Burroughs, язык программирования Алгол. Эксплуатируется в Канадском департаменте по делам занятости и иммиграции.

Система SPANAM (SPANish-AMERICAN, США). Опирается на принципы, разработанные в рамках системы GAT (см. табл. 11.1): перевод осуществляется прямым способом; грамматики «встроены» в программное обеспечение; анализ локальный. Общий порядок обработки текста аналогичен тому, который описан ниже для систем серии SYSTRAN. Программное обеспечение основано на более современной технологии; имеет модульную организацию. Система реализована на базе основной модели ЭВМ фирмы IBM; язык программирования — ПЛ/1. Текст может вводиться через оптические читающие устройства. К словарям (содержащим по 54 000 единиц для каждого языка) имеется прямой доступ. Хотя качество перевода невысокое, вкуче со средствами экранного редактирования и форматирования система позволяет сократить затраты на перевод (по оценкам 1980 г.) на 61 %. Применяется тремя переводческими Панамериканской организации здравоохранения, переводящими по 4—6 тыс. слов в день каждый.

Системы серии SYSTRAN (System of TRANslation, США). По имеющимся сведениям (возможно, устаревшим), все системы строго бинарны, используют прямой подход к переводу и ориентированы на узкие тематические области. Обработка текста включает словарный поиск, выделение идиоматических оборотов, снятие омонимии частей речи, определение границ именных и глагольных групп, придаточных, соединительных конструкций, поиск подлежащего и сказуемого, перевод предлогов, перевод многозначных слов, морфологический синтез, позиционные преобразования.

Эти процедуры объединены в несколько циклов, при выполнении каждого из которых весь текст просматривается подряд; коррекция получаемых результатов по данным более поздних циклов обработки не предусматривается. Грамматика «встраивается» в программное обеспечение.

В последнее время сообщается о переходе к модульному построению систем.

Большое значение придается полноте словарей и тщательному контролю правильности включаемых в них синтаксических и семантических сведений. Помимо словарей общей лексики, используются тематические глоссарии. В англо-французской системе входной и выходной словари содержат более 100 000 единиц каждый. В русско-английской системе русский словарь включает 150 000 общепотребительных единиц и 160 000 терминов, английский — около 400 000 единиц. Серия реализована на основной модели ЭВМ фирмы IBM и на ЭВМ Siemens 7740; программы написаны большей частью на ассемблере. Основные пользователи: BBC США, НАСА, Комиссия европейских сообществ, фирмы General Motors, Херох и др. Невысокое качество перевода компенсируется выигрышем в скорости (до 300 000 страниц в час, если не считать постредактирования). Стоимость перевода — 8 дол. за страницу, стоимость составления и ввода одной словарной статьи при пополнении словарей — 4 дол.

Система TAUM-METEO (Канада). Явилась побочным результатом работ по проекту TAUM (Traduction Automatique, Université de Montreal), разрабатывавшемуся с 1965 по 1981 гг. Это пока единственная в мире полностью автоматическая система МП: на ее вход подается весь поток метеосводок, поступающих в общегосударственную сеть метеорологического оповещения Канады, система выделяет в нем англоязычные сообщения, переводит их на французский язык и подает переводы в общую сеть без предварительного просмотра человеком. Система строго специализирована (рассчитана только на данный тип текстов). Перевод осуществляется прямым способом и сопровождается автоматическим выявлением ошибок. При их наличии соответствующие тексты (5—10 % от всех обрабатываемых) передаются для перевода человеку. Система работает на ЭВМ CDC Cyber 7600. Программное обеспечение написано на Q-языке, лингвистическое — на особом метаязыке высокого уровня, компиляторы и интерпретаторы для этого метаязыка — на Паскале. В день переводится 24 000 слов; требуемое для этого время работы центрального процессора — 15 мин.

Система TAURAS (Япония). Перевод осуществляется в два этапа: этап анализа, обеспечивающий представление входного предложения в виде структуры «семантических» (падежных) зависимостей, и объединенный этап межъязыковых операций и синтеза. Для ускорения анализа предусматриваемые им синтаксические и семантические процедуры отделены друг от друга, взаимодействуя только через общие для них структуры данных. Синтаксический анализ многовариантен и опирается на формализм РСП-грамматик. Для семантического анализа разработан особый аппарат — грамматики лексических сетей переходов. Их правила (позволяющие задавать преобразования деревьев и переходить от синтаксических структур к «семантическим» с одновременным снятием имеющейся в первых структурах неоднозначности) привязаны к конкретным лексемам и задаются в специальном словаре, работающем с использованием предкомпиляции. Пред- и постредактирование текста, может вестись параллельно с самим процессом его перевода, для чего вся информация о ходе АП, которая может требоваться редактору, записывается на так называемую «доску», к которой имеют доступ и система АП, и система редактирования. Работа той и другой системы синхронизируется методом «семафоров».

Система TITRAN (Title TRANslation, Япония). Особенности системы во многом определяются структурной ограниченностью языка заглавий японских научно-технических статей, для перевода которых она предназначена. Перевод осуществляется в два этапа: этап сегментации и морфологического анализа исходного заголовка (с одновременной проверкой каждой пары соседних слов на грамматическую совместимость) и этап прямого перевода полученной цепочки морфологических представлений слов. Второй этап включает три основных цик-

ла: обработку цепочек существительных, допускающих пословный перевод с сохранением исходного словоупотребления; обработку именных групп без придаточных (с вставкой при переводе предлогов и других служебных слов) и обработку придаточных (с выбором требуемых вместо них в переводе герундиальных, инфинитивных, причастных и других конструкций). По специальным алгоритмам обрабатываются единицы, не найденные в словаре, отрицание, знаки препинания, сочинительные конструкции, уточняется позиция в тексте перевода определений и обстоятельств и т. д. Работа системы почти полностью сводится к детерминированным процедурам, что способствует ее эффективности. Система реализована на ЭВМ FACOM M200, язык программирования ПЛ/1. При ее экспериментальной проверке в 1984 г. на материале 3379 заглавий 86 % из них было переведено правильно (если не считать артиклей).

Система TITUS (Франция). Многоязычная система с лексическими и синтаксическими ограничениями на язык входных текстов — рефератов работ по вопросам текстильной промышленности. Рефераты составляются в интерактивном режиме, и, если референт нарушает предусмотренные ограничения, система информирует его о требуемых ею изменениях вводимого текста. Начало придаточных помечается при вводе особыми символами. При анализе формируется особое промежуточное представление, с которого может происходить перевод на любой из четырех языков системы.

Комплекс систем АНРАП (СССР). См. § 12.2.

Система КНР-1* (КНР)¹. Осуществляет перевод заглавий научно-технических статей. Используется синтаксический и логико-семантический анализ. Перевод выдается только для заголовков, которым система может дать однозначную интерпретацию (50—60 % от их общего количества). Если хотя бы один элемент допускает неоднозначное понимание, заголовок передается для коррекции или нового перевода человеку. Система реализована на ЭВМ IBM 370. Словари содержат до 50 000 единиц. Программное обеспечение разработано Китайской компанией математического обеспечения. К 1990 г. намечено перенести систему на микроЭВМ и разработать (также на базе микроЭВМ) японско-английскую систему.

Система КНР-2* (КНР). Переводятся в диалоговом режиме тексты по металлургии, поступающие на вход на магнитных носителях. Переводы после редактирования подаются в систему автоматического набора для издания. Программное обеспечение разработано теми же силами, что и для системы КНР-1.

Система СИЛОД (СССР). Обеспечивает сигнальный информативный перевод, лексический перевод-подстрочник и технический лексический перевод. Состоит из программного комплекса и комплекса автоматических словарей. Программный комплекс включает универсальные программы анализа, опирающиеся на словари словоформ с заданной структурой словарной статьи, универсальные программы морфологического синтеза, основанные на единой структуре статей выходных словарей, и ряд программ, специфических для отдельных языков, например программы морфологического анализа для русского, французского и турецкого языков.

Входные словари имеют объемы (на начало 1987 г.): английский — 18 000 слов и 12 000 оборотов (словосочетаний), французский — 10 000 слов и 4000 оборотов, испанский — 8000 слов и 3000 оборотов, китайский — 6000 слов и 13 000 оборотов, японский и турецкий — по 1000 слов и по 500 оборотов, русский — 4500 основ и 1000 оборотов. Выходные словари есть для русского и английского языков; это словари основ с полной морфологической информацией. Русский выходной словарь (МАРС — Многоцелевой Автоматический Русский Словарь) содержит 50 000 основ, английский — 3500 основ. Отдельные модули

¹ Здесь и далее * означает, что системе дано условное название по наименованию организации-разработчика или поставщика, поскольку в имеющейся литературе она описывается без специального наименования.

имеются для семантических и тезаурусных описаний. Отсылки к ним даются в статьях МАРС, через который обеспечивается семантическая обработка для всех языковых пар. Словари ориентированы на вычислительную технику и на публицистику. Система реализована на ЭВМ ЕС-1022 (ОС ЕС) в диалоговом и пакетном режимах. Языки программирования — ПЛ/1, Ассемблер. Для функционирования программного комплекса ведения словарей достаточно 120 Кбайт оперативной памяти. Входной словарь при объеме 150 000 единиц занимает 3 Мбайт, выходной русский словарь — 4 Мбайт. Тексты вводятся без редактирования; перевод, хотя бы дефектный, выдается для любого входа.

Система ЧПИ (СССР). Опирается на микрословари по узким тематикам: химия полимеров (объем словаря — 22 000 единиц), органический катализ, физическая химия, охрана окружающей среды. Используется пословно-пооборотный принцип МП, т. е. снятие омонимии и некоторые грамматические преобразования при переводе обеспечиваются включением соответствующих словосочетаний (оборотов) в словари. Отмечается, однако, что, будучи приемлемым для англо- и французско-русского МП, этот подход неэффективен, в частности, для русско-казахского перевода. Система работает в диалоговом режиме. Для экранного интер- и постредактирования разработаны специальные функции, выполняющие замену, вставку, удаление и перестановку отдельных слов. В качестве базовой используется система разделения времени ОЛ ЕС ЭВМ, позволяющая вести редактирование одновременно на 10 дисплеях. Средняя скорость работы одного редактора в таком режиме — 7 страниц в час.

11.3. Развивающиеся системы машинного перевода

Развивающимися здесь названы те из разрабатываемых в настоящее время систем МП, которые предназначены непосредственно для промышленного использования, но к середине 1986 г. еще не достигли этого уровня. Таковы системы ATHENE, EUROTRA, GETA, Mu, NARA, SCANLAN, SHALT, SMU, TAS-1, Thalia-3, TRANSOFT, АПАЧ, РУСЛАН, из отечественных разработок — ПАРС, ПЕРЕВОДЧИК, СИМПАР, ФРАП и ЯРАП. Многие из этих систем реализуют «трех-этапный» подход к процессу перевода, требующий раздельного осуществления анализа входного текста, трансфера и синтеза текста перевода. Однако в ряде систем предусматривается вместо этого прямая стратегия перевода или сочетаются элементы того и другого подходов. Есть также системы ЧМП и системы, предназначенные для чисто вспомогательных операций (например, TAS-1). Существенно варьируются системы данной группы и по другим параметрам: по используемым в них типам формальных средств, по способу организации словарей и грамматик и т. д.

Ниже все названные системы перечисляются в алфавитном порядке, и для каждой из них отмечаются основные особенности организации ее математического и лингвистического обеспечения, а также характеризуется (при наличии соответствующих сведений) степень ее общей разработанности и ожидаемый уровень качества перевода.

Система ATHENE/E (Automatic Translator of Hitachi for English and Nihongo with Editing support/English, Япония). Осуществляет пофразный перевод с использованием межязыковых операций. При анализе, помимо обычных КСГ, применяются перечни недопустимых последовательностей синтаксических классов слов (для снятия омонимии частей речи), семантико-синтаксические модели управления предложными группами (заданные для английских глаголов, существительных и прилагательных), сведения о дистрибуции и семантической сочетаемости предлогов и т. п. Результатом являются расширенные НС-структуры входных предложений. В ходе межязыковых операций они дополняются недостающей падежной информацией и преобразуются в расширенные падежные структуры. На этапе синтеза уточняются залого и времена выходных глаголов, обрабатываются случаи несовпадения частей речи входных и выходных слов, вводятся служебные слова и выполняются позиционные преобразования. Если

слово имеет неоднозначный перевод, на дисплей выводятся все имеющиеся варианты. При выборе пользователем одного из них автоматически корректируются окончания спрягаемых форм и служебные элементы словосочетаний. Для ведения и пополнения словарей предусмотрены специальные человеко-машинные интерфейсы.

Система ATHENE/N (расшифровку ATHENE см. выше; конечное N означает Nihongo — японский язык). Как и в системе ATHENE/E, перевод осуществляется пофразно через этап межъязыковых операций. Анализ начинается с сегментации и морфологической обработки входного текста; далее для каждого простого предложения выявляются отношения синтаксического подчинения (так называемые отношения «какари-укэ», буквально «обслуживающее лицо — клиент») и падежные отношения между составляющими его словами и строится «структура понятийных зависимостей». Используется грамматика, записанная на специальном проблемно-ориентированном языке, падежные модели управления глаголов и прилагательных, сведения о семантической сочетаемости слов, общие ограничения на структуру зависимостей (типа проективности), а также эвристические правила выбора предпочтительных вариантов анализа (например, при определении границ придаточных), основанные на статистике. Грамматика включает набор подграмматик, различающихся способом и местом их применения в ходе анализа, типом описываемых языковых явлений и характером правил. Язык для записи правил содержит средства идентификации и преобразования подграфов размеченного ориентированного графа; реализован он с помощью интерпретатора. Порядок применения правил может контролироваться с помощью специальных параметров в их составе.

Межъязыковые операции обеспечивают, в частности, обработку семантико-синтаксических категорий и глубинных падежей, не совпадающих во входном и выходном языках (например, замену категории «становления» на категорию «причинности»).

Для синтеза полученная структура зависимостей преобразуется в НС-структуру с одновременной заменой в ней понятийных категорий (типа «причинности») нужными выходными словами. Пример перевода: Софутосуэ но кино: тэ-сүто но тамэ но тэсүто ко: моку о кэйто-тэки ни сакусэй суру AGENT гихо: о тэйан сита («Предложена методика AGENT, обеспечивающая систематическое порождение тестируемых элементов для функциональной проверки программного обеспечения») ⇒ AGENT technique that generates test item for software function test systematically was proposed.

Система EUROTRA (ЕЭС). Создается в рамках самого крупного на западе проекта МП. Задумана как многоязычная система, охватывающая основные языки ЕЭС. Перевод должен быть пофразным и происходить через межъязыковые операции. Предложения должны представляться в виде деревьев зависимостей с пометами типа «признак — значение признака». Грамматика строится, как в системе GETA (см. ниже): это набор подграмматик, содержащих неструктурные правила подстановки; интерпретатором служит система подстановок общего типа и согласованный с ней язык управления грамматиками-процессами.

Каждая страна-участница проекта должна создать модули анализа и синтеза для своего языка. Межъязыковые модули разрабатываются центральной группой и должны сводиться к минимуму — в идеале только к лексическим заменам. Для согласования разных описаний задано небольшое число признаков, которые должны порождаться при анализе любого языка. Для конкретных языков могут добавляться и другие признаки, но в модулях анализа их рекомендуется ограничивать. Программное обеспечение полностью отделено от лингвистического. В отличие от всех других систем здесь оно не определяет даже общую стратегию основных этапов обработки текстов: ее смогут задавать лингвисты на специальном управляющем языке, запись на котором будет компилироваться в программу, интерпретирующую «статические» правила, описывающие факты ЕЯ. Пока неясно, достижима ли требуемая для этого гибкость программного обеспечения.

Система GETA (Groupe d'Etudes pour la Traduction Automatique, Франция). Представляет собой интерактивную систему, по замыслу являющуюся многоязычной. Использует межязыковые операции на уровне деревьев зависимости, включающих как «глубинную», так и некоторую «поверхностную» информацию о входных предложениях.

Имеется три основных типа программных средств различной мощности: для преобразования линейных цепочек в деревья (например, при морфологическом анализе), для преобразований типа дерево-дерево (при синтаксическом анализе и межязыковых операциях) и для преобразования деревьев в линейные цепочки (например, при морфологическом синтезе). Каждое из них служит интерпретатором для лингвистических данных, записанных на соответствующем метаязыке. Обработка текста разбита на ряд последовательных этапов, для каждого из которых выбрано одно из этих средств. Принцип выбора: мощность используемых средств не должна превышать ту, которая минимально необходима для выполнения задач данного этапа. Это повышает быстродействие системы и дисциплинирует разработчиков лингвистических описаний. Грамматика системы организована в виде сети, в которой грамматика высшего уровня образует граф управления относительно подграмматик, непосредственно осуществляющих лингвистические операции. Особый алгоритм предусмотрен для работы с межязыковым словарем.

Теоретически программное обеспечение системы допускает реализацию любых лингвистических моделей, но написано оно на языках программирования низкого уровня, что лимитирует эти возможности. В рамках французского Национального проекта по МП, за основу которого принята эта система, планируется перепрограммировать ее на Лиспе и довести до коммерческого уровня. Ее методология воплощается также в ряде других систем, например в проекте EUROTRA.

Система Му (Япония). Ориентирована на язык аннотаций научно-технических статей. Перевод осуществляется пофразно. Выделяется семь его этапов: анализ — построение для входного предложения дерева его глубинно-падежных зависимостей (допускающего совмещение разных вариантов его структурной интерпретации, а также информации разных лингвистических уровней); замена конструкций, специфических для входного языка, более универсальными; межязыковые операции: преобразование полученной структуры в структуру, в большей мере отражающую специфику выходного языка; синтез дерева составляющих выходного предложения; структурные трансформации в целях улучшения стилистических характеристик выходного текста; морфологический синтез.

Обработка текста на большинстве этапов ведется под управлением словаря, содержащего богатую морфологическую, синтаксическую, семантическую, тезаурусную и другую информацию. Грамматические описания задаются средствами специализированного формального аппарата, позволяющего определять любые преобразования деревьев, а также организовывать грамматику как множество подграмматик, различающихся типом описываемых языковых явлений, и устанавливать порядок их применения в зависимости от выполнения определенных условий. Реализован он на языке программирования Лисп (версия UTILISP для ЭВМ FACOM M382), а также Symbolics 3600; объем системной программы — 10 000 строк.

К 1986 г. словарь системы включал 74 000 японских существительных и 5000 слов других частей речи. Грамматика, применяемая при анализе, содержала 900 правил, распределенных по 60 подграмматикам; грамматики, работающие на этапах межязыковых операций и синтеза, насчитывали около 800 правил. Система опробовалась в режиме японско-английского АП на материале 10 000 фраз. При оценке качества переводов для 1700 из них понятность была признана приемлемой примерно в 73,2 % случаев, адекватность — в 63,6 % случаев.

Система NARA (Япония). Перевод осуществляется через этап межязыковых операций, на который вынесены, в частности, процедуры снятия неоднозначности. Грамматическое описание каждого языка опирается на формализм НСГ

и состоит из ряда независимых компонентов, описывающих соответственно морфологию данного языка, выделенный для него набор бинарных синтагм, порядок слов, синтаксис и семантику. В процедурной части системы каждому компоненту сопоставлен отдельный модуль. К 1986 г. создана система-прототип, работающая в режиме «он-лайн» в реальном масштабе времени.

Система SCANLAN (ПНР). Предназначена для пофразного перевода рефератов по химии, биохимии и проблемам коррозии. Основные этапы обработки текста: графематический анализ (выделяющий в исходной фразе формулы, латинские вкрапления и т. п.), морфологический анализ, синтаксический анализ с выделением конструкций, требующих преобразования при переводе, выполнение требуемых преобразований путем обращения к специальному контрастивному словарю конструкций, замена исходных лексических единиц их переводными эквивалентами, морфологический синтез, форматирование выходного текста с учетом заданной ширины страницы и полей. Словарь входного русского языка включает словари общеупотребительной лексики (5000 словоформ), словарь общих и специальных сокращений (600 единиц) и словарь оборотов (120 единиц). Словарь польских переводных эквивалентов содержит 1200 основ.

Система SHALT (Япония). Предназначена для перевода руководств по программированию. Перевод осуществляется пофразно с использованием межъязыковых операций на уровне расширенной НС-структуры. Анализ является чисто синтаксическим, организован в направлении снизу вверх и допускает многовариантность. Варианты структур упорядочиваются по степени их «правдоподобия». Для грамматически неправильных фраз строятся частичные структуры. На этапе межъязыковых операций исходная структура обходится вначале сверху вниз, и конструкции, специфические для входного английского языка, заменяются в ней конструкциями, имеющими прямые японские соответствия. Затем полученная структура обходится снизу вверх, и по ней строится расширенная НС-структура выходного японского предложения.

Выбор переводных эквивалентов происходит на уровне «синтаксических», а не «семантических» падежей (это считается более эффективным), но опирается на семантические проверки, одновременно служащие целям устранения лексической и структурной неоднозначности. Используются 24 семантических признака, отражающие специфику предметной области.

В ходе синтеза уточняется перевод некоторых английских служебных слов, вводятся недостающие японские служебные единицы, устраняются, где необходимо, кореферентные повторы и определяется нужный порядок слов. На каждом из этапов анализа, межъязыковых операций и синтеза работает свой словарный массив. Программное обеспечение системы написано на Лиспе.

Система SMU* (Saint Malaysia University, Малайзия). Это система ЧМП, опирающаяся на логико-алгоритмический аппарат системы GETA (см. выше). Использует языки программирования разных уровней: от Ассемблера до ПЛ/1. Основные этапы процесса перевода: морфологический анализ (на основе грамматики объемом 90 правил), синтаксический анализ (300 правил), лексические межъязыковые операции, структурные межъязыковые операции (47 правил), синтаксический синтез (120 правил), морфологический синтез (120 правил). Наибольшую проблему составляет разрешение неоднозначности глаголов и предлогов и создание словарей. К 1986 г. построен входной английский словарь (объемом 5000 единиц), выходной малайский словарь (4000 единиц) и словарь англо-малайских межъязыковых соответствий. Параллельно готовится лексическая база данных для различных информационных задач.

Система TAS-1 (Translator's Assistant System, ГДР). Фактически сводится к средствам экранного редактирования и к двум автоматическим словарям в помощь переводчику: двуязычному словарю терминов и одноязычному толковому словарю, позволяющим выдавать на экран переводные эквиваленты и/или словарные толкования отдельных слов. Обрабатываются только те слова входного текста, которым пользователь приписал при редактировании специальные пометы; если эти слова стоят не в словарной форме, он должен пометать в них

также начало отделяемой части. К 1986 г. разработаны основы программного обеспечения системы. Пока оно написано на псевдокоде, при машинной реализации предполагается использовать языки Бейсик или Паскаль.

Система Thalia-3 (Translator as Helpful And Linguistic Assistant, Япония). Планируется несколько режимов работы: от прямого пословного перевода до перевода через язык-посредник с учетом сверхфразовой структуры текста. К 1986 г. реализован пофразный перевод с использованием межъязыковых операций на уровне так называемой ADS-структуры — графа зависимостей, в вершинах которого стоят «понятия» (словозначения), а их характеристики и отношения между ними заданы в терминах формализма фреймов. На каждом из этапов анализа, межъязыковых операций и синтеза используется свой тип словарей и грамматик и свой формальный язык для записи грамматических правил.

Анализ и синтез опираются на КС-грамматики, расширенные процедурами обработки валентностей (при анализе) и указаниями на требуемый порядок слов и (или) требуемые типы зависимостей (при синтезе). Для описания межъязыковых операций разработан более мощный формализм, позволяющий задавать любые преобразования ADS-структуры. Для упрощения отладки и совершенствования лингвистического обеспечения предусматривается модульная организация каждого правила.

В словарях дается богатая морфологическая, синтаксическая и семантическая информация (включая надежные модели управления). Словари общей и терминологической лексики отделены друг от друга. Объем словарей общей лексики планируется довести до 60 000 единиц.

Система реализована на ЭВМ серии MELCOM-COSMO. Имеются средства экранного редактирования перевода (с выдачей на дисплей в случае неоднозначности всех допустимых вариантов), средства редактирования словарей и грамматик (предусмотрен специализированный язык программирования высокого уровня) и консультативная система, диагностирующая состояние процесса АП и информирующая о нем пользователя.

Система TRANSOFT (США — ФРГ). Использует прямую стратегию перевода. Обработка текста ведется под управлением таблиц: вначале по таблице синтагм осуществляется синтаксический анализ входной фразы и перестановка составляющих ее слов в порядке следования их эквивалентов в переводе; затем по словарной таблице выбираются сами требуемые эквиваленты. Табличная информация к каждой словарной единице (словоформе или словосочетанию) включает признак ее синтаксического класса, основной переводной эквивалент и, возможно, ряд альтернативных переводов, которые могут выбираться по контексту. Синтагмы имеют вид цепочек синтаксических классов, для каждого из которых указано, должна ли соответствующая ему единица учитываться при последующем анализе (тем самым обеспечивается «свертывание» синтагмы к представляющим ее единицам) и в какую позицию в пределах синтагмы ее следует переместить (представляющие единицы обычно оставляются для дальнейшей обработки в исходном порядке). Система реализована на мини-ЭВМ PDP-11/70 в ОС M11+ (язык программирования ANS MUMPS). Осуществлен экспериментальный перевод учебника по медицине (с немецкого языка) и ряда более коротких текстов (с других языков). Средняя скорость перевода превышала 9000 слов в час.

Система АПАЧ (СССР). Обработываемый тип текстов — аннотации статей и другие материалы по вопросам изготовления и эксплуатации насосов. Перевод осуществляется через этап межъязыковых операций. Теоретико-лингвистическую основу системы составляет функциональная порождающая модель языка, разработанная в 60-х годах пражскими лингвистами. Математическое обеспечение опирается на формализм Q-систем [Colmerauer, 1971]. Система находится на полужэкспериментальной стадии разработки; планируется ее промышленное использование в концерне Sigma.

Система ПАРС (Перевод Английских Рефератов и Статей, СССР). Ориентирована на тексты по порошковой металлургии; допускает адаптацию к любой узкой предметной области. Процесс перевода включает этап построения по

исходному тексту его русского подстрочника и три этапа обработки подстрочника: устранение омографии (вариантности переводных эквивалентов исходных слов) с учетом соседних слов текста; перестановку стоящих подряд существительных; морфологический синтез. На первом этапе работает словарь английских словоформ, не содержащий никакой информации, кроме их русских эквивалентов, на следующих этапах — словарь русских эквивалентов, где указаны основы соответствующих русских слов, их части речи, парадигмы словоизменения, модели грамматического управления и предметные области, к которым они относятся. Система реализована на ЕС ЭВМ (ОС 6.1 и выше, язык программирования — ПЛ/1). Имеются сервисные программы создания и корректировки словарей. Объем английского словаря на конец 1987 г. составляет 7000 единиц. Далее систему планируется дополнить блоком семантико-синтаксического анализа по методу «клише» (см. § 11.4 о системе ИПКИР), а также блоком сверхфразового анализа.

Система ПЕРЕВОДЧИК (СССР). Предназначена для нужд бытового общения туристов. Опирается на двужычные словари основ и оборотов словосочетаний) и грамматические описания входных и выходных языков в терминах так называемых базисных структур. Машинная форма грамматики — это система иерархически организованных таблиц решений. Алгоритмический компонент системы включает модули морфологического, лексико-фразеологического и (семантико-) синтаксического анализа и синтеза, последовательность работы которых зависит от типа входного и выходного языков. Экспериментальная версия системы реализована на микроЭВМ ЕС 1040 и опробована на материале более 6000 высказываний для каждого из трех языков: английского, французского, русского.

Система РУСЛАН (ЧССР). Предназначена для перевода с чешского на русский язык руководств по эксплуатации ОС ЕС ЭВМ, разрабатываемых в ЧССР. Математическое и лингвистическое обеспечение строится на основе того же общего подхода, что и в системе АПАЧ (см. выше). Для семантико-синтаксического анализа чешских текстов используется подсистема, созданная в рамках вопросно-ответной системы ТИВАК. К концу 1986 г. была реализована первая экспериментальная версия системы РУСЛАН. На конец 1987 г. блоки анализа и синтеза разработаны практически полностью, словарь содержит 5000 единиц.

Система СИМПАР (Система Интерактивного Машинного Перевода с Английского языка на Русский, СССР). Ориентирована на подязык нефтехимии и термодинамики. Состоит из ряда совмещаемых модулей, последовательно подключаемых к базовой схеме, ядром которой является автоматический словарь. Имеется 15 сервисных программ АС, программа-диспетчер и четыре модуля синтаксического и семантического анализа-синтеза. Для описания синтаксиса применяется модифицированный вариант НС-грамматики. Семантическая обработка опирается на индексы лексико-семантических групп и типов лексико-семантической сочетаемости, приписанные в АС словам входного языка. Система реализована на ЭВМ ЕС 1040; языки программирования — ПЛ/1 и Ассемблер.

Система ФРАП (СССР). См. § 12.4.

Система ЯРАП (Японско-Русский Автоматический Перевод, СССР). Планируется два режима работы: получение нормализованного русского подстрочника японских текстов и пофразный перевод с использованием межъязыковых операций на уровне семантико-синтаксических структур зависимостей.

Первый режим предусматривает сегментацию и морфологическую обработку исходного текста, первичное снятие омонимии путем проверки соседних слов на грамматическую совместимость, переход к русским эквивалентам исходных единиц и изменение порядка их следования («нормализацию» подстрочника) с учетом самых общих соответствий между японским и русским словопорядком.

Для работы системы во втором режиме в ней предполагается воплотить основные методологические принципы, выработанные в связи с исследованиями по системе АРАП [Шалыпина, 1974; 1977]: лексикографический способ лингвистического описания (построение описаний и лексических и грамматических еди-

ниц в виде «словарных статей» единой структуры, использующих единую символику и единые форматы записи и различающихся только степенью общности); последовательно валентностное описание всей языковой синтагматики, от морфологии до сверхфразового синтаксиса; осуществление основных процедур обработки текстов под управлением словаря; возможность одновременного учета информации разных языковых уровней. Сам процесс обработки, в отличие от системы АРАП, должен быть в основном одновариантным, с возможностью возвратов и переходов к альтернативным вариантам, что должно контролироваться с помощью динамических оценок «перспективности» рабочего варианта, основанных на статистике.

К 1987 г. реализована небольшая модельная версия начальных блоков системы и проведены эксперименты по получению подстроки для текстов по нефтехимии. Программное обеспечение написано на Ассемблере ЕС ЭВМ и использует ряд программных модулей комплекса АНРАП (см. § 12.2). Для ввода японских текстов в ВЦП ГКНТ и АН СССР создается система меню, которая позволит вводить иероглифы и знаки японских азбук пользователям, не знакомым с японской или китайской письменностью.

11.4. Экспериментальные системы машинного перевода

В данном параграфе рассматриваются те из современных систем МП, которые разработаны (или разрабатываются) в основном как аппарат для экспериментальных лингвистических исследований, без непосредственной ориентации на промышленное применение. Системы, предполагающие такое применение, описываются здесь в том случае, если они еще не вышли на соответствующий уровень, а реализуемые в их рамках подходы к МП не служили ранее объектом широко-масштабных экспериментальных исследований. К экспериментальным системам можно отнести, например, DLT, HIT, KBMT, LUTE, MOPTRANS, NTT-CSK, PERSIS, Rosetta, SALAT, SEMSYN, SUSY, TIT-SUT, TRANSLATOR, UMIST, UOP, отечественные системы ИПКИР, ФР-П, ЭТАП и др. Ниже кратко характеризуются основные принципы построения лингвистического и математического обеспечения каждой из названных систем и указывается, насколько это возможно в пределах настоящего изложения, специфика реализуемых в их рамках научных концепций. В конце параграфа сообщается также о ряде новых экспериментальных систем, сведения о которых появились в печати к осени 1988 г.

Система DLT (Distributed Language Translation, Нидерланды). Используется промежуточное представление входных текстов на языке-посреднике (ЯП) — модифицированном экспертно. Предусматривается интерредактирование на стадии анализа (для снятия различных видов неоднозначности) и полностью автоматический синтез. Синтаксический анализ имеет традиционный характер. Семантический анализ, диалог с интерредактором (осуществляемый под управлением меню) и переход к ЯП опираются на лексико-семантическую экспертную систему SWESIL (Semantic Word Expert System for the Intermediate Language). Это недетерминированная система, упорядочивающая варианты анализа по степени их семантической приемлемости. В основе ее лежит лексическая база знаний, где каждому слову приписана вся релевантная для него информация — лингвистическая (морфологическая, синтаксическая и т. п.) и экстралингвистическая (характеризующая понятия и их отношения в «модели мира»). Каждый модуль анализа имеет независимый доступ к используемым им компонентам этой базы и только к ним, так что вся обработка идет под ее управлением. На 1987 г. объем базы знаний составлял 2000 единиц, объемы словарей для перехода от английского языка к ЯП и от ЯП к французскому языку — по 4000 единиц. К 1990 г. их планируется довести соответственно до 3000, 10 000 и 8000 единиц. Ожидается, что система сможет работать в режиме самообучения, а также применяться при компьютерном обучении иностранным языкам.

Система HIT (Harbin Institute of Technology, КНР). Ориентирована на перевод заглавий научно-технических статей. При анализе используется «метод

аналогий». Эксперименты 1981 г. показали более высокое качество перевода, чем для системы КНР-1 (см. § 11.2).

Система KBMT (Knowledge Based Machine Translation, США). Перевод осуществляется через «каноническое смысловое представление», опирающееся на формализм фреймов. Исходное описание включает как лингвистические, так и экстралингвистические сведения («знания» по отдельным предметным областям). Используются формализм ЛФГ и «сущностно-ориентированный формализм» [Hayes, 1984], позволяющий объединять все типы сведений, релевантных для некоторой единицы, в единый фрагмент описания, что упрощает их верификацию. Перед началом работы описание подвергается многоэтапной предкомпиляции и преобразуется вначале в расширенную НСГ, а затем в особую табличную форму. Механизм предкомпиляции и вид получаемых таблиц зависят от требуемого типа процедур (анализ или синтез). Текст обрабатывается строго слева направо, и при вводе его с дисплея анализ может начинаться прежде, чем будет введена до конца хотя бы одна фраза. Сообщается, что эффективность алгоритма анализа выше, чем у других алгоритмов, основанных на КС-грамматиках, и не зависит от объема используемой таблицы. Для обработки грамматически неправильных входных фраз разработана методика постепенного снижения жесткости учитываемых при анализе ограничений. Реализована экспериментальная англо-японская версия системы, ориентированная на область общения «врач — пациент» (язык программирования Лисп). Ведутся исследования по испанскому, французскому, немецкому и итальянскому языкам.

Система LUTE (Language Understander, Translator, Editor, Япония). Перевод осуществляется через промежуточное представление текста, включающее два компонента: «смысловое» представление предложений, имеющее вид расширенной падежной структуры, и «речевую память», где накапливается информация о более крупных отрезках текста. Анализ начинается с морфологической обработки предложения и построения его гипотетической синтаксической структуры путем применения к нему синтаксических «структурных моделей» в направлении сверху вниз. Затем происходит интерпретация каждого фрагмента этой структуры с помощью семантических фреймов и рекурсивная интеграция их в единую «смысловую» структуру, осуществляемая в направлении снизу вверх и сопровождаемая оценкой построенной части структуры с точки зрения ее синтаксической и семантической удачности. В зависимости от полученной оценки возможен возврат к более раннему этапу анализа и построение альтернативной структуры. Межязыковые операции выполняются путем рекурсивного обхода обрабатываемой структуры сверху вниз.

Лингвистическое обеспечение, помимо синтаксических описаний (задаваемых средствами НСГ) и (синтактико-) семантической информации (падежных моделей управления слов, сведений о синонимических, причинно-следственных и других «концептуальных» отношениях и т. п.), опирающейся на формализм фреймов, включает также «метазнания», определяющие способы интеграции фреймов в ходе анализа и формируемые, в свою очередь, в терминах фреймов. Любая информация в зависимости от степени ее общности может быть помещена либо в общую «базу знаний», либо в словарь.

На 1986 г. словари обоих языков системы насчитывали по 3000 единиц. Система реализована на базе Symbolics Lisp-machine, язык программирования Zeta-Lisp. Объем программного обеспечения — 850 Кбайт, объем словарей — 4 Мбайт. Имеется мощная подсистема редактирования фреймовых описаний, средства слежения за работой программ и другие сервисные средства.

Система MOPTRANS (Memory-Organization-Packets-based TRANslator, США). Используется язык-посредник — структура «концептуальных зависимостей», образуемая весьма общими смысловыми понятиями и отношениями. Анализ текста, в отличие от стандартного подхода к этой процедуре, опирается в первую очередь на семантическую обработку текста, а синтаксические сведения привлекаются уже для выбора между разными семантически допустимыми вариантами формируемой структуры. Основу системы составляют два типа знаний: «пакетные» и «абстрагирующие». Пакетные знания задают причинно-

следственные, временные и другие последовательности абстрактных событий, общие для многих различных ситуаций (этим они отличаются от «сценариев», имеющих более частный характер). Абстрагирующие знания — это, например, сведения о родовидовых отношениях между понятиями. Снятие лексической неоднозначности обеспечивается не проверкой сочетаемости лексем, а переходом от них через абстрагирующие знания к обобщенным понятиям и последующим использованием имеющихся для этих понятий пакетных знаний. Правила работы со знаниями обоих типов не зависят от конкретного языка.

Система NTT-CSK* (Nippon Telegraph and Telephone public corporation, CSK research institute, Япония). Перевод осуществляется пофразно с использованием межъязыковых операций на уровне так называемых *f*-структур, опирающихся на формализм ЛФГ. *f*-структура предложения — это иерархия функций, соответствующих морфологическим, синтаксическим, падежным и другим лингвистическим категориям и имеющих своими аргументами и значениями слова и составляющие данного предложения. При анализе вначале выявляется НС-структура предложения, затем по данным словарей и грамматик ее компонентам приписываются релевантные для них «лексические функции» и на их основе строятся системы «функциональных уравнений». Решение их (с применением операций унификации) дает связную *f*-структуру всего предложения. Далее для ее элементов по сопоставленным им межъязыковым правилам строятся функциональные уравнения, задающие частичные структуры выходного предложения, и по тому же алгоритму, что и при анализе, формируется его полная *f*-структура, поступающая на этап синтеза. Подчеркивается двусторонний (обратимый) характер словаря, используемого для межъязыковых операций: в него входят единицы и входного, и выходного языков, имеющие только отсылки к правилам межъязыковых преобразований; сами правила вынесены в отдельный массив.

Система PERSIS (PERsian analySIS, Япония). Разработана прежде всего система анализа, строящая для предложения сеть зависимостей на основе грамматики продукции и словаря «понятий» (словозначений). Грамматика содержит 850 синтаксических продукций, использующих 42 структурных признака и задающих, в частности, глагольные модели управления (их выделено около 150). Словарные статьи «понятий» и структурные представления предложений или их компонентов, формируемые при анализе, имеют одинаковый формат: это всякий раз один из 17 стандартных наборов признаков, где значением отдельного признака может быть синтаксическая или семантическая характеристика описываемой единицы, указание на ее (возможную или реальную) роль в предложении и т. д. или же еще один («вставленный») набор признаков, релевантный для другого слова (например, синтаксически ей подчиненного). В ходе анализа используется около 100 правил определения применимости продукции (задающих тип согласования признаков, указанных в составе продукции, с признаками единиц, имеющихся в контексте) и около 150 правил интеграции признаков (объединяющих признаки тех единиц, к которым применяется некоторая продукция, в информацию к образуемой при этом более крупной единице). Анализ одновариантен и предусматривает возможность возвратов. Программное обеспечение написано на Лиспе для системы с разделением времени NEC ACOS-100. К 1986 г. проведен эксперимент по анализу и построению английского подстрочника для 200 персидских фраз (1600 слов). Успешные результаты получены в 88,5 % случаев.

Система Rosetta (Нидерланды). Язык описывается в терминах грамматик Монтегю, т. е. путем определения множества элементарных выражений с их грамматическими категориями и множества синтаксических правил, позволяющих выводить категории неэлементарных выражений из категорий их компонентов. Каждому синтаксическому правилу сопоставляется логическое, определяющее способ выведения значения выражений, соответствующих исходному правилу, из значений составляющих их более мелких выражений. Дерево такого логического вывода предполагается использовать как промежуточное представление языковых выражений при их переводе. В силу обратимости грамматик Мон-

тею одни и те же грамматические описания должны быть применимы и при анализе, и при синтезе.

Система SALAT (ФРГ). Перевод предполагается осуществлять через язык-посредник с использованием базы знаний и правил логического вывода. Трактовка языка-посредника близка к той, которая была принята в системе CETA (см. табл. 11.1): универсален только его структурно-синтаксический компонент, базирующийся на логико-семантических основаниях (в частности, на работах Монтегю); лексические единицы обрабатываются с помощью словаря межязыковых соответствий. Основные этапы перевода: морфологический и синтаксический анализ входного текста с получением его поверхностной структуры; преобразование поверхностной структуры в глубинную (опирающуюся, как и сведения, хранящиеся в базе знаний, на аппарат логических формул); снятие лексической и синтаксической неоднозначности путем обращения к базе знаний; переход к глубинной структуре выходного текста (кроме двуязычного словаря, здесь работают правила логического вывода, обеспечивающие структурные преобразования); построение поверхностной структуры выходного текста и текста как такового.

Система SEMSYN (ФРГ). На вход поступают семантические сети с абстрактным синтаксисом и английской лексикой, получаемые на этапе межязыковых операций в японско-английской системе ATLAS-II (см. § 11.2). При их обработке формируется представление, где вся структурная информация задана в виде схем фреймового типа: «падежных схем» для предикатов, «концептуальных схем» для имен, «реляционных схем» для описания квантификации, модальности, сочинения и т. п. По нему строится функционально-грамматическое представление немецкого предложения, содержащее уже немецкую лексику (оно сочетает в себе элементы структуры составляющих и структуры зависимостей), и далее синтезируется окончательное немецкое предложение. Основным средством организации лингвистической информации служит база знаний, реализованная на основе системы FLAVOR в рамках ЛИСП-формализма. В ней даются соответствия «семантических» (английских) единиц немецким лексемам, указываются способы конкретизации исходных общих понятий по контексту (с учетом имеющейся терминологической базы), способы выявления недостающих грамматических сведений (например, о числе существительных и о требуемых ими артиклях) и т. д. Система опробовалась на материале 2000 заглавий японских статей по информатике и технологии программирования. При оценке качества переводов для 400 из них все переводы признаны приемлемыми, в том числе 68 % — полностью адекватными, 16,7 % — полностью грамматически правильными.

Система SUSY (ФРГ). Перевод осуществляется строго пофразно с использованием межязыковых операций на уровне глубинных структур зависимостей, имеющих абстрактный синтаксис. Предусматривается богатая словарная информация — как морфолого-синтаксическая, так и лексико-семантическая (в том числе сведения о лексических валентностях). В соответствии с преимущественно экспериментальной направленностью системы обработка текста может вестись различными средствами: с применением КС-грамматик или трансформационных правил; под управлением словарей, правил или таблиц; на основе информации признакового, категориального или структурного типа; с учетом или без учета валентностей (служащих большей частью для снятия различных видов неоднозначности и для обработки неполных или нестандартных входных структур). Последовательность отдельных уровней анализа задана жестко и не допускает возвратов. Однако это компенсируется многовариантностью соответствующих процедур и использованием различных эвристик, весовых оценок, упорядочения правил по степени предпочтительности и т. п. Лингвистическое и программное обеспечение в последней версии системы последовательно разделены. Ведется разработка сервисных средств. Система реализована на ЭВМ TR440 (языки программирования Фортран и Ассемблер). Изучаются возможности сопряжения ее с системой TITRAN (см. § 11.2).

Система TIT-SUT* (Tokyo Institute of Technology, Science University of Tokyo, Япония). Перевод осуществляется в два этапа: этап грамматического

анализа и этап, совмещающий семантический анализ, межъязыковые операции и синтез. Словарь и грамматика первого этапа, а также общий программа построения «семантической» (падежной) структуры фразы и выбора переводных эквивалентов слов, опирающаяся на их словарные описания, не зависят от выходного языка и могут работать в многоязычной системе. В семантическом словаре для слова перечисляются семантические ограничения на заполнение валентностей («слотов») в его различных значениях и выбираемые с учетом этого переводные эквиваленты. Перечисление организовано «от общего к частному», так что ограничения, общие для нескольких значений слова, проверяются только один раз. Выполнение взаимосвязанных семантических проверок контролируется с помощью механизма «демонов». В ходе межъязыковых преобразований и синтеза учитывается не только падежная, но и поверхностно-синтаксическая структура входных предложений. Основные формальные средства — расширенный LINGOL и специальная система гибкого описания семантики SRL. Система рассчитана на большие ЭВМ; экспериментальная версия реализована на микроЭВМ фирмы Оки Дэнки Kōgē CP/M и фирмы Фудзичу M-11 (языки программирования соответственно Stiff Upper Lisp и LISP-86).

Система TRANSLATOR (США). Перевод осуществляется через язык-посредник. Анализ предусматривает многовариантную морфолого-синтаксическую и семантическую обработку входных текстов. На уровне ЯП работает особый модуль «Инспектор» — род экспертной системы, которая моделирует действия человека-интерредактора, выбирая наиболее правдоподобный из альтернативных вариантов анализа с учетом имеющейся базы знаний. Лексика ЯП задается в виде соответствий между единицами входного (входных) и выходного (выходных) языков. Словарные статьи в словаре ЯП, как и в словарях, используемых при переходе от входных языков к ЯП и от ЯП к выходным языкам, содержат сведения об отношениях типа род-вид, часть-целое и т. п., о подязыке, к которому относится данная единица, и о ее падежных валентностях. В грамматике ЯП определяются прежде всего временные, пространственные и другие обстоятельственные отношения. База знаний сочетает в себе «кратковременную память», куда помещаются сведения о содержании анализируемого текста, и «долговременную память», хранящую информацию, релевантную для разных текстов по общей тематике (например, информацию о соответствующем подязыке).

Система UMIST (Великобритания). Процесс перевода включает этапы анализа, межъязыковых операций и синтеза, на каждом из которых работает свой тип словарей и грамматик. Основные формальные инструменты — ЛФГ и обобщенная НСГ. В переводе по возможности сохраняется вся та неоднозначность входного текста, которую не удастся снять при анализе. Для ее отражения в промежуточном представлении текста применяются формализм граф-схем, особые признаки, выделяющие разные типы многоязычных слов, и другие средства. Предусматривается интерактивное пред- и интерредактирование, при котором пользователю указывается, как следует переформулировать входный текст, чтобы для него мог быть получен удовлетворительный перевод. Диалог с пользователем ведется только в терминах входного языка, т. е. система рассчитана в основном на перевод с родного языка пользователя непосредственно в ходе составления им того или иного текста с применением персональной ЭВМ. Для разработки и совершенствования системы имеются сервисные средства различной мощности, ориентированные соответственно на конечного пользователя, на лингвиста-составителя конкретных описаний, на системного лингвиста и на проектировщика системы. Система реализована на ЭВМ ICL PERQ, язык программирования Пролог.

Система UOP* (University of Osaka Prefecture, Япония). При разработке применяются специальные средства автоматизации программирования L-MAPS (Library Module Aided Program Synthesizing) — программа на языке Лисп объемом около 1000 строк, позволяющая писать спецификации программ на ограниченном японском или английском языке. Эти спецификации автоматически приводятся к формальному виду, конкретизируются путем обращения к библиотечным модулям и развертываются в программы на заданном языке программи-

рования. Одновременно для них порождается комментарий на том же ограниченном естественном языке. Переход от формальных выражений к естественно-языковым и обратно основывается на логике предикатов 1-го и частично 2-го порядка и организован как этап межъязыковых операций при АП. При этом используются падежная грамматика и промежуточное представление обрабатываемых выражений в терминах универсальных субфреймов предикатных слов.

Экспериментальная англо-японская версия системы УОР включает 24 прикладных и 30 базовых модулей. Объем ее программного обеспечения, порожденного системой L-MAPS, — около 1000 строк на языках Си и FRANZ LISP. Межъязыковые операции происходят на том же уровне, что и в самой системе L-MAPS. Процедуры анализа, первоначально написанные на расширенном LINGOL, в версии 1986 г. опираются на логические операции унификации и субституции. Лингвистическое описание задается правилами подстановки, работающими с фреймами.

Система ИПКИР* (СССР). Основным компонентом является словарь так называемых клише — полных английских предложений и их русских переводов с указанием порядка следования входящих в каждое предложение словосформ, их семантических признаков и того, какие из них синтаксически связаны (тип и направление связей не учитываются). Кроме него имеется англо-русский терминологический словарь (ТС), единицы которого снабжены морфологическими и семантическими пометами, словарь английских служебных слов и блок синтаксической коррекции. При обработке входного предложения оно членится на сегменты по словарю служебных слов, и для каждого сегмента ищутся клише, содержащие те же словосформы в том же порядке. При полном лексическом совпадении перевод клише выдается в качестве перевода сегмента. Если некоторая словосформа сегмента лексически отличается от соответствующей словосформы найденного клише, она переводится с помощью ТС, а согласование ее перевода с переводом других единиц клише выполняет блок синтаксической коррекции. При неоднозначном выборе клише возможен учет информации о соседних предложениях текста. К 1981 г. ТС содержал 10 000 единиц, словарь клише — 2000 заголовков статей по электротехнике. Проведен ряд ручных экспериментов, давших положительные результаты.

Система ФР-II (СССР). Перевод осуществляется через этап межъязыковых операций. Используется многовариантный анализ фильтрового типа, доходящий до уровня поверхностно-синтаксической структуры. При разработке учитывался опыт построения и опробования системы ФР-I, разрабатывавшейся с 1954 г. и функционировавшей в качестве экспериментальной модели с 1956 г. Система была реализована на ЭВМ БЭСМ-4; ее подробное описание см. в [Кулагина, 1979].

Комплекс ЭТАП (СССР). См. § 12.3.

Приведенные выше описания отражают разработки и исследования в области МП по состоянию на начало 1987 г. К 1988 г. появились материалы, прежде всего [MT Summit, 1987; IC MT, 1988; COLING, 1988], которые не могли быть учтены здесь в полной мере. Отметим только наиболее очевидные из тех тенденций, выводы о которых могут быть сделаны по этим материалам.

1. Продолжает расширяться география исследований по МП. По последним данным, в работу по созданию систем МП, помимо тех стран, которые упомянуты в табл. 11.1 (Великобритания, ГДР, Гонконг, Малайзия, Нидерланды, Канада, КНР, ПНР, СССР, США, Франция, ФРГ, ЧССР, Япония), включились также Боливия (многоязычная система ATAMIR, используемая в Бюро переводов Комиссии Панамского канала и в Международном центре переводов фирмы Wang), Индия (руско-тамилская система TUMTS, разработанная в 1983 г. в Тамильском университете), Таиланд (система, базирующаяся на программном обеспечении ARIANE-78, созданном во Франции группой GETA), Норвегия (англо-норвежская система ENTRА, созданная в Бергенском университете), Швеция (многоязычная система SWETRA, разрабатываемая в университете г. Лунд). В исследовании отдельных вопросов, связанных с задачами МП, и в разработках боль-

ших систем МП, осуществляемых различными международными организациями или транснациональными корпорациями, принимают участие и исследователи многих других стран, в том числе Бельгии, ВНР, Израиля, Индонезии, Испании, НРБ, Финляндии, Швейцарии и др.

Заметно оживление и расширение работ в области МП в тех центрах, где такие работы ведутся уже давно. Так, в Саарском университете (ФРГ) начали разрабатываться французско-немецкая система ASCOF и многоязычная система MARIS, в Манчестерском университете (Великобритания) — англо-японская система NTRAN, в Монреальском университете (Канада) — англо-французская система CRITTER. Имеются сообщения о новых промышленных и экспериментальных разработках по МП в Японии. Среди них выделяется многоязычный проект ODA (Overseas Development Agency), сравнимый по многим параметрам с европейским проектом EUROTRA. В финансировании проекта ODA принимают участие ведущие японские компании, однако большую часть расходов по его выполнению берет на себя правительство.

2. Расширяется также круг языков, для которых ставится в экспериментальном или промышленном плане задача МП. Так, в рамках латиноамериканской системы ATAMIRI, уже используемой для промышленного перевода с английского языка на испанский, планируется осуществлять также перевод с английского на немецкий, нидерландский, французский, шведский и итальянский языки и с испанского на португальский, венгерский и аймара (язык андских индейцев, который используется также в качестве языка-посредника).

Индийская система TUMTS служит (пока в экспериментальном режиме) для перевода текстов по астрономии с русского языка на тамильский.

Система, разрабатываемая в Таиланде, предназначена для перевода с английского на тайский язык.

Система SWETRA, разрабатываемая для перевода со шведского и на шведский язык, опробовалась, помимо английского, французского и немецкого, также на материале грузинского и самоанского языков.

Японский проект ODA предусматривает перевод с японского на малайский, индонезийский, тайский и китайский языки.

3. При разработке новых систем МП продолжают использоваться и совершенствоваться самые разные из существующих к настоящему моменту подходов к решению лингвистических проблем перевода.

В системе TUMTS принят прямой способ перевода. В системах, разрабатываемых в Малайзии, Индонезии и Таиланде, а также в системах ASCOF, CRITTER и некоторых других процедура перевода является многоуровневой и включает этап межъязыковых операций, который, в свою очередь, может быть многоуровневым (система ASCOF). В системе ATAMIRI предусмотрен язык-посредник, в качестве которого выбран индийский язык аймара. Концепция языка-посредника реализуется также в рамках проекта ODA.

Для описания естественного языка и процедур обработки текстов широко используются уже апробированные модели. Выше говорилось о применении в англо-тайской системе алгоритмического комплекса ARIANE-78. В системе NTRAN применяется формализм расширенных НСГ. В основу системы CRITTER, существенно опирающейся на опыт создания систем серии TAUM (см. табл. 11.1), положен аппарат древесных предикатно-аргументных структур и их преобразований. В системах ASCOF и MARIS развиваются методы, исследованные в связи с разработкой систем GETA и SUSY (см. табл. 11.1).

Вместе с тем создаются и проверяются на различном языковом материале новые виды формализмов и алгоритмических средств. Например, в системе SWETRA используется так называемая референтная грамматика, представляющая собой специальный вариант обобщенной НСГ. Во многих японских системах предусматривается использование ряда средств и методов искусственного интеллекта. В проекте ODA планируется обработка текста не только на уровне морфологии и синтаксиса предложения, но и на уровне структуры абзаца, которая до сих пор в системах МП последовательно не эксплицировалась.

Несмотря на то, что системы МП приобретают все большее промышленное и коммерческое значение, основные проблемы МП, включая выбор оптимального подхода к общей организации процесса перевода, отнюдь нельзя считать окончательно решенными. Напротив, в настоящее время эти проблемы становятся объектом все более интенсивных фундаментальных исследований.

Глава 12.

Отечественные системы машинного перевода

А. Д. Бакулов, Н. Н. Леонтьева, З. М. Шалыпина

12.1. Три подхода к решению задач машинного перевода

Отечественные системы МП ориентированы в большинстве своем на полностью автоматический перевод с той долей постредактирования, которая обусловлена качеством выходного текста МП. Однако возникающие при этом задачи решаются в разных системах разными средствами. В лингвистическом отношении здесь можно выделить три основных подхода, особенности которых наиболее четко выявляются при сопоставлении таких разработок, как комплексы АНРАП и ЭТАП и система ФРАП.

Комплекс АНРАП в большей степени, чем две другие системы, учитывает существующие в настоящее время условия промышленной эксплуатации отечественных систем МП и наиболее продвинут в технологическом отношении: во Всесоюзном центре переводов научно-технической литературы и документации ГКНТ и АН СССР (ВЦП) он включен в качестве одного из компонентов в специально организованную технологическую цепочку, обеспечивающую обработку текстов с момента их поступления от заказчика и до момента передачи заказчику отредактированных переводов этих текстов. Лингвистическое и программное обеспечение комплекса также ориентировано на максимальную технологичность: используется лексикографический способ организации лингвистической информации, повышающий надежность работы комплекса (благодаря этому он может работать с любыми типами входных текстов, в частности с грамматически неправильными предложениями и словосочетаниями, и давать какие-то — хотя бы дефектные — переводы даже при наличии во входном тексте слов, отсутствующих в используемом словаре); обработка текстов ведется в рамках прямого подхода к переводу, позволяющего увеличить скорость всего процесса; целям повышения скорости и надежности обработки текста служит и ряд принятых в рамках комплекса АНРАП более частных решений: например, способ представления всего множества омонимов, сохраняющихся для той или иной единицы текста на каждом шаге его обработки, в виде единой морфолого-синтаксической сущности — так называемого многоязычного слова. Заметим, что, как показано в [Whitelock et al., 1986] при рассмотрении аналогичного способа представления омонимов, выбранного для системы UMIST, в его пользу могут быть приведены и определенные теоретические аргументы. Однако качество первичных (не прошедших постредактирования) переводов, обеспечиваемых данным комплексом, невысокое и требуемые объемы постредактирования весьма велики, так что наиболее эффективен он лишь в качестве средства оперативного сбора информации.

Комплекс ЭТАП и система ФРАП еще не вышли на промышленный уровень. Комплекс ЭТАП представляет собой классический вариант «грамматической» системы МП. Он основан на широком использовании механизма межъязыковых операций и обладает развитым синтаксическим компонентом, обогащенным семантическими сведениями. Для записи и обработки соответствующих сведений создан мощный логико-алгоритмический аппарат, предоставляющий лингвистам-

разработчикам системы хорошие экспериментальные возможности. Тексты, покрываемые имеющимися в комплексе словарями и грамматиками, получают при его использовании переводы достаточно высокого качества. Однако возможности работы с грамматически неправильными входными предложениями или с неполным покрытием их словарями в комплексе ЭТАП (особенно в системе ЭТАП-1) представлены слабее, чем в комплексе АНРАП и в системе ФРАП.

В системе ФРАП принят подход, отличный и от того, который реализован в комплексе АНРАП, и от того, на который опирается комплекс ЭТАП. Данная система проектировалась с самого начала как система с семантическим компонентом, где перевод должен осуществляться через род языка-посредника. Синтаксический компонент в ней может не давать на выходе правильного дерева анализа. После него предусматривается этап семантической интерпретации, на вход которого могут подаваться структуры любого уровня полноты и правильности. Именно за счет работы этого этапа должно обеспечиваться высокое качество перевода; на этом же этапе должны уточняться первоначально неправильные, неоднозначные или неполные входные структуры. В логико-алгоритмическом отношении особенность системы ФРАП, отличающая ее от комплекса ЭТАП, состоит в том, что синтаксическая информация задается и декларативными, как в указанном комплексе, и процедурными средствами.

Каждая из трех существенно различных стратегий построения систем МП, демонстрируемых тремя названными системами, имеет и свои достоинства, и свои недостатки. Можно предполагать, что дальнейшее развитие и совершенствование систем любого из этих типов должно будет идти по пути их постепенного сближения, с использованием опыта, накопленного в рамках каждого из альтернативных подходов.

Ниже все три системы рассматриваются более подробно.

12.2. Комплекс систем англо-русского и немецко-русского автоматического перевода АНРАП

Комплекс предназначен для использования в крупных переводческих организациях и должен обеспечивать перевод на русский язык англо- и немецкоязычной политематической документации. Входная информация может быть любой формы; в частности, она может поступать из банков данных или иметь вид массивов, записанных на магнитных лентах. Предусмотрена возможность интер- и постредактирования результатов перевода в интерактивном режиме, а также возможность оперативной коррекции и пополнения информационных массивов в ходе их эксплуатации. Комплекс строится на базе двух ранее созданных систем АП: АМПАР (англо-русской) и НЕРПА (немецко-русской). Интеграция их в единый комплекс обусловлена использованием в обеих системах общего программно-технологического обеспечения, общего русского словаря и общей подсистемы русского морфологического синтеза. Программное обеспечение комплекса написано на языке Ассемблер ЕС ЭВМ, операционная система СМВ.

Из двух систем, положенных в основу комплекса, ведущее место принадлежит системе АМПАР. Это первая из отечественных систем АП, получивших промышленное значение. Она начала создаваться еще в конце 50-х годов как развитие пионерских разработок И. К. Бельской [Бельская, 1969]. Руководили ее созданием вначале Ю. А. Моторин, затем Ю. Н. Марчук (см. общее описание системы в [Марчук, 1983]); современная версия программного обеспечения системы разработана под руководством Б. Д. Тихомирова [Тихомиров, 1984]. В 1979 г. система была принята в опытно-промышленную эксплуатацию, в 1981 г. — в промышленную. К осени 1987 г., помимо ВЦП, она была установлена также в пяти отраслевых организациях. Скорость ее работы (без учета затрат времени на подготовку и ввод переводимых текстов, а также на постредактирование полученных переводов) составляет 3—5 авторских листов в час на ЕС-1035. Словари этой системы, по данным 1987 г., насчитывают около 40 000 английских и около 50 000 русских словарных статей. Емкость требуемой для системы машинной памяти — 20 Мбайт.

Система НЕРПА принята в промышленную эксплуатацию в 1985 г., Объем немецкой части словаря к осени 1987 г. составлял 30 000 словарных статей, по остальным параметрам система НЕРПА аналогична системе АМΠΑР.

Рассматривая содержательные аспекты организации и функционирования комплекса АНРАП, необходимо разграничивать, с одной стороны, те принципиальные возможности и ограничения, которые заданы для допустимых в его рамках лингвистических описаний общей логикой его математического и программного обеспечения («встроены» в это обеспечение), и, с другой стороны, конкретные способы реализации имеющихся принципиальных возможностей в ныне действующих версиях лингвистического обеспечения.

«Встроенные» принципы описания языка и обработки текстов в комплексе АНРАП сводятся к следующему.

1. Лингвистическое описание организуется как совокупность словарей и грамматик. Синтагматические и трансформационные сведения о языке могут указываться в тех и других в виде соответственно общих и частных «грамматических схем», так что противопоставление лексических и грамматических описаний может в принципе идти только по степени общности включаемой в них информации при том, что характер и способ ее записи в описаниях обоих типов может быть одним и тем же.

2. Лингвистическая информация синтагматического и трансформационного характера представляется в процедурной форме. Для ее записи разработан специальный проблемно-ориентированный язык программирования — так называемый язык стандартных операторов, в терминах которого могут компактно задаваться такие операции над обрабатываемым текстом, как выбор в нем единиц с определенными признаками в качестве «опорных точек» для дальнейшей обработки, поиск в их контексте единиц, занимающих в этом контексте определенную позицию и имеющих определенные признаки, приписывание найденным единицам новых признаков, элиминация имеющихся у них признаков или перенос признаков от одной единицы к другой, введение в текст ранее отсутствовавших в нем единиц, изменение порядка следования имеющихся единиц, их элиминация и т. д.

3. Обработка текста на всех этапах анализа, собственно перевода и синтеза осуществляется циклически. Каждый цикл состоит в просмотре всего обрабатываемого текста подряд слева направо и в применении к нему процедур одного определенного типа. При этом может использоваться любая информация, полученная для единиц текста в рамках всех предыдущих циклов его обработки, так что возможность выделения таких циклов по существу представляет собой механизм, обеспечивающий процедурную реализацию идей уровневого описания естественного языка.

4. Применение при обработке текста общеграмматического компонента лингвистического описания происходит под управлением так называемой ведущей программы — общей управляющей программы (монитора), которая задает порядок обращения к отдельным общеграмматическим схемам (подграмматикам), организует тем самым сеть этих схем (подграмматик). Работа ведущей программы не зависит от состава конкретного обрабатываемого текста. Обращение к очередной схеме соответствует началу очередного цикла обработки, в течение которого в тексте при его последовательном просмотре ищутся единицы, способные служить «опорными точками» для применения данной схемы; всякий раз, когда обнаруживается такая единица, схема применяется, а затем просмотр текста продолжается дальше.

5. В отличие от схем общеграмматического типа частные грамматические схемы, описывающие индивидуальные свойства отдельных лексических единиц или их лексико-семантических групп, применяются под управлением словаря и текста. Поэтому то, какие схемы и в какой последовательности вызываются при обработке конкретного текста, определяется наличием и порядком следования в нем соответствующих лексических единиц. Обеспечивается это следующим образом: при просмотре текста в рамках того или иного «частнограмматического» цикла его обработки для каждого очередного слова проверяется, сопоставлена ли ему в том отделе словаря, который должен участвовать в работе данного

цикла, какая-либо грамматическая схема. При наличии такого указания инициируется работа этой схемы, и после того, как она проработает, происходит переход к следующему слову текста. Если при общеграмматической обработке текста в ходе каждого цикла к разным словам применяется одна и та же (и только одна) схема, то здесь для лексически различных слов в пределах одного и того же цикла могут работать разные схемы. Повтор в тексте одних и тех же слов приводит к повторному вызову релевантных для них схем.

6. Общий механизм обработки текста не привязан формальным образом к понятию предложения. С одной стороны, для получения перевода фразы не требуется в обязательном порядке предварительного построения ее полной синтаксической структуры: в частных случаях каждое словосочетание или даже каждое слово может переводиться отдельно. Тем самым обеспечивается устойчивость работы системы: переводной текст, хотя бы весьма несовершенный, может выдаваться даже при значительных сбоях в ее грамматическом компоненте. С другой стороны, анализ контекста той или иной единицы не ограничен пределами предложения или даже абзаца: любые ограничения этого рода зависят только от содержательных лингвистических соображений и должны задаваться эксплицитно в соответствующих грамматических схемах.

7. Промежуточное представление информации об обрабатываемом тексте, формируемое и преобразуемое в ходе его анализа, собственно перевода и синтеза, имеет вид линейной цепочки так называемых информационных ячеек, которые сопоставляются идентифицируемым в ходе обработки лингвистическим объектам. В каждой ячейке хранится (в форме соответствующих признаков) вся накапливаемая для соответствующего объекта информация. На характер самих выделяемых объектов никаких априорных ограничений не налагается: информационные ячейки могут формироваться как для слов или словосочетаний, реально присутствующих в исходном и (или) выходном тексте, так и для любых лингвистических конструктов, которые могли бы вводиться на каких-то этапах его обработки, например, для составляющих. В принципе этими средствами можно задать любую древесную структуру.

Однако возможности отображения в таком представлении языковой неоднозначности ограничены. Лексическая и лексико-грамматическая омография фиксируется в нем путем формирования общей информационной ячейки для всего множества омографичных друг другу единиц, представляемого тем самым как единое «многозначное слово»; информация о нем в такой ячейке должна заменяться затем в ходе устранения омографии на информацию о том из омографов, который выбран для конкретного обрабатываемого контекста. Для представления синтаксической неоднозначности используется метод приписывания той или иной синтаксической функции (например, функции подлежащего) всем единицам, которые могли бы выполнять данную функцию в альтернативных вариантах анализа обрабатываемого контекста. По мере устранения неоднозначности эта информация для тех слов, для которых она оказывается ложной, стирается или заменяется другой. Таким образом, в каждый данный момент обработки текста его представление как бы одновариантно, хотя по ходу такой обработки один вариант может заменяться другим. Это дает выигрыш в скорости и устойчивости функционирования системы, но осложняет работу по корректировке и совершенствованию ее лингвистического обеспечения (из-за плохой обзорности и проверяемости соответствующей информации).

В современной версии лингвистического обеспечения системы АМПАР описанные общие возможности используются следующим образом.

Лексическая («словарно-управляемая») информация распределена по шести словарным массивам, каждый из которых работает в рамках отдельного цикла обработки текста. Это: 1) английский морфологический словарь, задающий, в частности, классификацию слов, способных принадлежать в разных значениях к разным частям речи, по типам возможных здесь альтернатив; 2) словарь фразеологических оборотов; 3) переводной словарь однозначных слов; 4) переводной словарь многозначных слов (содержит для каждого из них набор процедурно представленных правил — так называемую схему перевода многозначного слова.

с помощью которой для описываемого слова выбирается требуемый в контексте вариант его перевода); 5) словарь перевода многозначных слов по умолчанию; 6) словарь русских слов с морфологической, словообразовательной и синтаксической информацией (типа сведений о падеже управляемого слова).

К общеграмматической (схемно-управляемой) информации относятся схемы обработки конструкций, выделяемых по наличию в них слов с определенными частями речи, морфологическими показателями (типа окончания *ing* у английских глаголов) и другими частотными грамматическими характеристиками. С помощью общеграмматических схем анализируются также слова, не найденные в словаре, и т. п.

Разбиение общей процедуры обработки текста на циклы определяется не столько независимыми лингвистическими соображениями, сколько степенью доступности для системы на каждом этапе ее работы той или иной необходимой для перевода информации. Хотя в целом в системе прослеживается разграничение трех основных этапов перевода — анализа, собственно перевода и синтеза, оно не носит принципиального характера. Так, морфологический и семантико-синтаксический анализ входного английского текста во всех случаях, где это в принципе возможно, производится сразу в категориях выходного (русского) языка. Для многозначных слов значительная часть анализа вынесена на этап выбора их переводных эквивалентов, а для конструкций типа вторичной предикации, специфичных для английского языка и требующих при переводе сложных синтаксических преобразований, — на этап синтеза соответствующих русских конструкций.

Последнее частично связано с тем, что промежуточное представление структуры словосочетаний и предложений строится в существующей версии системы АМПАР в терминах частей речи и членов предложения (синтаксических функций слов). Ни границы составляющих, ни отношения зависимостей в нем в явном виде не фиксируются, и при необходимости учета такой информации она должна всякий раз заново извлекаться из обрабатываемого контекста посредством его повторного анализа.

В лингвистическом обеспечении системы НЕРПА предусматривается несколько большая независимость процедур анализа и синтеза. В частности, грамматическая информация об исходных немецких словоформах представляется в их информационных ячейках отдельно от соответствующей информации об их русских эквивалентах. Ряд других отличий (таких, как создание отдельных таблиц немецких морфологических показателей, выделение особого цикла обработки сложных существительных, особого цикла анализа и преобразования рамочных конструкций и т. п.) обусловлен в основном спецификой немецкого языка по сравнению с английским.

Эксплуатация комплекса АНРАП предполагает, помимо работы систем машинного перевода как такового, осуществление определенных вспомогательных процедур. Это прежде всего процедуры подготовки текстов к переводу, т. е. отбор текстов по заданной тематике и их предредактирование (обработка нетекстовых включений, корректура типографских дефектов и разметка текста для набивки), а также процедуры интер- и постредактирования, оформления конечного продукта перевода и т. д. Комплекс содержит программно-технологические средства, позволяющие по заявкам организаций создавать из базового комплекса упрощенные системы перевода (версии исходной системы), адаптированные к тематическим потребностям и техническим возможностям этих организаций. При этом пользователи системы будут иметь возможность самостоятельно создавать и эксплуатировать в составе системы дополнительные словари, относящиеся к их тематической области (за исключением только словарей многозначных слов). Создание словарей будет доступно специалистам, не знакомым со спецификой комплекса АНРАП.

Предполагается, что ВЦП сможет не только создавать и распространять версии комплекса АНРАП, но и организовать кооперацию по сбору созданных тематических словарей, их обработке и распространению среди организаций-пользователей.

12.3. Системы семейства ЭТАП

12.3.1. Общие сведения о системах. Работа над комплексом ЭТАП (ЭлектроТехнический АП) ведется с 1974 г. В 1980 г. была сдана в экспериментальную эксплуатацию первая очередь (ЭТАП-1), осуществляющая перевод с французского языка на русский (описание см. в [Апресян и др., 1983]). В 1985 г. была сдана в опытно-промышленную эксплуатацию вторая очередь системы (ЭТАП-2), осуществляющая перевод с английского языка на русский. Система ЭТАП-2 была реализована на ЭВМ ЕС 1033 и работала в двух режимах: качественного перевода на основе полного синтаксического анализа фразы и пословного перевода, базирующегося только на морфологическом анализе. Второй режим использовался как аварийное средство, когда нельзя было построить синтаксическую структуру и перевод в полном режиме (подробное описание системы ЭТАП-2 см. в [Апресян и др., 1988]). Комплекс программ был написан на языке PL/1, затем для ускорения процесса перевода часть блоков была переписана на языке Ассемблер.

Объем словарей в реализованных версиях — 6000 входов на каждый язык в морфологическом словаре, из них 2000 описаны в комбинаторном словаре. На перевод заголовка в 8—10 слов (при сокращенном на 1/3 массиве правил) уходило около 1 мин. Время перевода одного предложения в 25—30 слов средней степени сложности составляло 3—4 мин. После 1985 г. комплекс ЭТАП программно не поддерживается (кроме режима морфологического перевода, включенного в некоторые действующие информационные системы ИНФОРМЭЛЕКТРО).

Системы ЭТАП-1 и ЭТАП-2 представляют собой классические СМП синтаксического поколения, состоящие из трех частей: анализ — преобразование (межязыковые операции, трансфер, собственно перевод) — синтез. В системе ЭТАП-1 реализован перевод через поверхностно-синтаксическую структуру. Систему ЭТАП-2, в которой перевод осуществляется на уровне нормализованных синтаксических структур, занимающих промежуточное положение между поверхностно-синтаксической и глубинно-синтаксической структурами, авторы квалифицируют как СМП 2,5 поколения [Апресян и др., 1988]. Уровень глубинно-синтаксических структур авторы считают достаточным для получения АП высокого качества, сравнимого с традиционным переводом.

В комплексе ЭТАП реализованы основные методологические установки модели «Смысл — Текст» [Мельчук, 1974]: 1 независимое описание входного и выходного языков. Каждая языковая модель располагает интегральным (единым) описанием морфологии, синтаксиса и словаря, которые полностью согласованы по типам помещаемой в них лингвистической информации, везде она записывается единообразно, на одних и тех же формальных языках; 2) декларативность задания лингвистических знаний, т. е. их полная независимость от алгоритмов; это позволяет использовать одни и те же грамматики при анализе и при синтезе; 3) неориентированность системы на одну ПО (кроме словаря: в словарях около половины единиц составляют термины, но сама область «электротехники» является достаточно широкой ПО).

Система ЭТАП-2 ориентирована на перевод связных текстов (статьи, рефераты) и заголовков патентов БД INPADOC. Каждая фраза получает единственный перевод, в большинстве случаев удовлетворительный. Неудовлетворительный перевод объясняется чаще всего отсутствием какой-то информации в словаре или (реже) лингвистическими недостатками в правилах. Вносимые изменения не нарушают дееспособности системы.

Перечислим и охарактеризуем основные этапы.

Этап *морфологического анализа* включает три блока: предморфологический анализ, выделение безусловных оборотов, собственно морфологический анализ.

Этап *синтаксического анализа* — центральный, он определяет лингвистическую стратегию системы (см. ниже более подробное описание).

Третий этап — *нормализация синтаксической структуры* английского предложения: снимаются те морфологические характеристики, которые свойственны

только английскому языку, приводятся к стандартному виду «слишком английские» конструкции. На этом же этапе осуществляется коррекция некоторых заведомо неправильных синтаксических структур с помощью правил «реинтерпретация». Это, по сути, лексико-семантические перестройки структуры.

Четвертый этап — *преобразование нормализованной английской структуры в нормализованную русскую*. Для этого морфологические характеристики английских слов заменяются соответствующими русскими морфологическими характеристиками, синтаксические конструкции английского языка преобразуются в соответствующие русские, английские лексемы заменяются их русскими эквивалентами. Иногда это простые преобразования, а иногда сложные, взаимозависимые.

Пятый этап — *развертывание нормализованной структуры* в синтаксическую структуру будущего русского предложения. Здесь рождаются все лексико-синтаксические своеобразия русского языка.

На долю шестого этапа — *синтаксического синтеза* — остаются две задачи: морфологизация синтаксической структуры и расстановка знаков препинания. Установление порядка слов и знаков пунктуации выполняется не глобальными правилами, а локальными.

Остановимся подробнее на лингвистическом обеспечении анализа.

12.3.2. Промежуточное представление. Синтаксическое представление в комплексе ЭТАП описывается деревом зависимостей, ребра которого помечены именами синтаксических отношений (СО), а узлами являются имена лексем предложения с наборами морфологических характеристик. В описании французского синтаксиса используется 50, в описании английского — 51, в описании русского — 42 СО.

12.3.3. Грамматика входного языка. Синтаксис входного языка задан *декларативно*, списком синтагм (правил). Они бинарны в том смысле, что каждая из них завершается инструкцией, предлагающей связать посредством СО два и только два слова фразы (которые обозначаются *X* и *Y*). Каждая синтагма описывает возможность (но не обязательность) построения некоторой синтаксической связи между словами при заданных условиях. Условия построения синтагмы для большей части списка заданы в терминах «древесного контекста», т. е. при условии наличия или отсутствия некоторых СО, связывающих *X* и/или *Y* с другими словами фразы. Условия синтагм разбиваются на группы (это делается лингвистом): необходимые/невозможные; линейный контекст/непосредственный древесный контекст/опосредованный древесный контекст. Корпус синтагм дает сугубо статическое описание языка, причем круговое, так как синтагмы ссылаются друг на друга и не могут быть упорядочены таким образом, чтобы автоматически задавался порядок их обработки при анализе. Это и означает полное отделение грамматики от алгоритма. Организация эффективной работы с правилами требует специальных алгоритмических приемов, не входящих в компетенцию лингвистов.

В системе ЭТАП-1 все правила преобразований, используемые на каком-либо этапе перевода, составляют единый последовательный файл, который полностью просматривается при обработке очередной фразы. В системе ЭТАП-2 все правила разбиваются на три группы: общие, частные и словарные. Общие правила, как и в системе ЭТАП-1, составляют последовательный файл. Частные правила (называемые также трафаретами) объединяются в индексный файл, доступ в который осуществляется по имени правила. Ссылки на правила из этого файла указываются в статьях комбинаторного словаря (КС) тех слов, для которых эти правила предназначены. Наконец, словарные правила записываются непосредственно в словарных статьях соответствующих лексем. Из 1000 правил грамматики 96 общие, 342 частные, остальные — словарные.

12.3.4. Словари. На стадии морфологического анализа используется морфологический словарь, на стадии синтаксического анализа — *комбинаторный словарь* (КС). Английский КС включает следующие типы информации о лексеме: 1) ее номер в КС; 2) часть речи; 3) тривиальный перевод на русский язык; 4) синтаксические признаки; 5) семантические признаки (дескрипторы); 6) модель представления лексемы, а именно сведения о количестве ее (семантических) актан-

тов, о способах их выражения и о требованиях к синтаксическим или семантическим свойствам ее потенциальных актантных зависимых; 7) имена трафаретных правил, необходимых для переработки фраз с данной лексемой, с указанием значений параметризованных переменных, если они есть в данном правиле; 8) словарные правила, необходимые для переработки фраз с данной лексемой. Русский КС включает аналогичные типы информации, за исключением перевода и семантических признаков лексемы.

В связи с реорганизацией грамматической информации в системе ЭТАП-2 КС тоже был дополнен отсылками к правилам: в словарной статье КС указывается или имя частного правила, или правило-трафарет (т. е. имя правила плюс значения тех параметров, при которых может быть реализовано частное правило), или правило целиком (словарное правило). Словарное задание правил перевода позволяет устранить буквализмы в переводе терминов, фразем и других несложных сочетаний, обеспечивает существенные для предметной области нетривиальные правила перевода.

Такое перераспределение компетенции грамматики и словаря в системе ЭТАП-2 по сравнению с предшествующей версией приближает комплекс ЭТАП к классу СМП преимущественно словарной ориентации, когда система производит самонастройку на перевод данной фразы.

12.3.5. Алгоритмы анализа и перевода. Алгоритм синтаксического анализа предполагает такую последовательность операций при обработке каждой фразы: 1) построение набора гипотетических СО; 2) определение вершины синтаксической структуры; 3) фильтры по окончательным и квазюкончательным СО; 4) проверка непосредственного древесного контекста; 5) работа правил предпочтения; 6) рассмотрение альтернативных деревьев; 7) проверка правильности синтаксической структуры; 8) аварийная организация связности графа.

Первым шагом алгоритма является построение заведомо избыточного графа гипотетических синтаксических связей между словами анализируемой фразы. Гипотетические связи возникают в результате работы с синтагмами, в которых на этом шаге проверяется лишь часть условий (условий на линейный контекст). В результате последующей проверки остальных условий синтагм и применения всевозможных фильтров из графа постепенно вычитаются лишние дуги и выделяется требуемое дерево.

На вход алгоритма перевода поступает первая построенная синтаксическая структура: предполагается, что грамматика устроена так, что эта структура всегда правильна. В системе ЭТАП-1 она могла оказаться дефектной ввиду отсутствия проверки правильности с точки зрения древесного контекста. В системе ЭТАП-2 она введена, поэтому, если первая построенная структура не удовлетворяет этому критерию, она отбрасывается и алгоритм возвращается к предыдущему варианту. В некоторых случаях все варианты разбора фразы могут завершиться неудачей — граф каждый раз после проведения чисток оказывается несвязным. Система не довольствуется выдачей пословного перевода, а проводит фиктивные связи от вершин фразы к несвязавшемуся узлу. На выходе всегда будет дерево зависимостей, среди дуг которого могут оказаться фиктивные. В этом случае перевод может содержать неточности, но другие фрагменты будут переведены адекватно.

12.4. Система французско-русского автоматического перевода ФРАП

12.4.1. Общие сведения о системе. Работы над системой ФРАП начались в 1975 г. Первая версия сдана в экспериментальную эксплуатацию в 1980 г., вторая версия — в 1985 г. При разработке модели системы была сделана попытка использовать готовые прототипы СМП второго поколения, однако имеющиеся модели были достаточно жестко привязаны к предметной области и не решали задачи, поставленной перед системой французско-русского перевода в условиях ВЦП: необходимость обеспечивать переводом относительно небольшой поток заказов при большом тематическом разбросе (практически любые

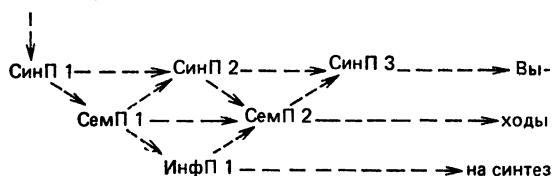


Рис. 12.1. Взаимодействие структур в системе ФРАП

области науки и техники). Система снабжена развитым семантическим компонентом, выполняющим функции стыковки с «семантикой» разных ПО и отделенным от синтаксического компонента, призванного обслуживать в равной мере любые ПО. Система ФРАП принадлежит к классу информационно-переводческих систем, осуществляющих перевод текста вместе с регулированием объема переводимой информации.

Существующая версия системы реализует часть модели и выполняет только переводческие функции, работая в трех режимах: подстрочник, грамматический режим, неполный семантический режим. Действующая версия системы выполняет реальные заказы на перевод, принимая на входе любые естественные тексты без тематических или иных ограничений и строя для них хоть какой-нибудь перевод на синтаксическом уровне. Этот перевод, опирающийся на основной морфологосинтаксический словарь (объемом 15 000 единиц для каждого языка), в котором всегда есть переводной эквивалент (либо единственный, либо «дежурный», т. е. самый общий), в дальнейшем может быть улучшен обращением в семантический компонент или в компонент ПО, обладающие более дифференцированным описанием тех же единиц. Если семантический словарь или тезаурус ПО ничего не предлагают для этой единицы, в окончательном переводе остается дежурный эквивалент, или, если единицы нет и в основном словаре, исходное французское слово. Такая устойчивость относительно входного материала и полноты словарей — свидетельство промышленных характеристик системы. Но система ФРАП остается экспериментальной системой, потому что она работает в неполном составе и пока не имеет промышленных характеристик по скорости, качеству, технологичности.

Система ФРАП была реализована на ЭВМ ЕС 1030 в операционной системе ДОС ЕС, в 1988 г. закончен ее перевод в операционную систему ОС ЕС (на ЭВМ ЕС 1055), имеющий целью оптимизацию программного и технологического обеспечения, а также доведение системы до промышленных параметров. В 1989 г. начата настройка словарей системы на политическую тематику, создание на ее основе компонента автоматического реферирования, переброска на технику IBM PC.

12.4.2. Теоретическая модель системы ФРАП и уровни контекстов. В основе системы лежит информационная модель понимания естественного текста [Леонтьева, 1985]. Она характеризуется возможностью построения глобальных структур для текста даже при неполноте и неоднозначности локальных анализов на разных промежуточных уровнях в основном за счет того, что соседние уровни могут обмениваться информацией. Анализ и синтез в информационной модели несимметричны. При анализе строится последовательность структур (представлений) текста, из них с точки зрения перевода существенны следующие:

- лексическое представление (ЛексП) текста;
- синтаксическое представление (СинП) предложения;
- семантическое представление (СемП) высказывания и целого текста (в единицах данного текста);

информационное представление (ИнфП) текста (в единицах воспринимающей системы; с точки зрения задачи перевода — в единицах той предметной области, которая привлечена к процессу понимания данного текста). На уровне ИнфП должна строиться и структура реферата.

Анализ в целом представляет собой прерывистый процесс (рис. 12.1), так как окончательная структура каждого уровня может быть достроена лишь после

взаимодействия со структурами высших уровней (или более глубокими структурами). В обозначениях рис. 12.1 СинП 1 — это состояние синтаксической структуры до ее взаимодействия с СемП; СинП 2 — синтаксическая структура, уточнившаяся в результате информации, полученной от СемП 1; СинП 3 — состояние СинП после получения информации от СемП 2, которая, в свою очередь, уточняется в результате взаимодействия с ИнфП.

В случае, если высшая инстанция ничего не добавляет или она вообще отсутствует, структуры переходят в следующее состояние на основе задаваемых извне критериев (свойств более правильной структуры данного уровня, правил предпочтения, веса), например: СинП 1 → СинП 2 → СинП 3.

На вход синтеза может поступить любая структура, будь она окончательная (т. е. при ее строительстве учтены все уровни компетенции системы и больше ничего добавить нельзя) или квазиоконечная, когда по каким-то соображениям вмешательство старших уровней нежелательно или затруднено, а все возможные операции данного уровня выполнены.

При пофразном переводе (через СинП) структуры высших уровней (СемП и ИнфП) не только уточняют структуру СинП, но и являются контекстом, влияющим на выбор переводных эквивалентов и структурных соответствий. Так, ЛексП задает линейный контекст, СинП — структурный (древесный) контекст в пределах предложений, СемП — смысловой контекст в пределах всего текста, ИнфП вводит в рассмотрение контекст ПО. Комбинации контекстов разных уровней — тоже контекст.

Чтобы контекст был полноценным, он должен представлять собой систему. Так, СемП может давать локальные интерпретации, но они могут быть ошибочными, если не поддерживаются контекстом целостной структуры. Перевод термина должен поддерживаться терминосистемой, стоящей за ИнфП.

Итак, каждый уровень имеет свои единицы и правила их комбинации (язык и грамматику), в терминах которых формулируются свойства правильной (идеальной) структуры данного уровня. Тем самым каждый уровень задается как отдельная система, которая может отвечать за определенный параметр текста перевода:

ЛексП отвечает за сохранение свойства непрерывности (каждая единица должна иметь один и только один перевод в тексте),

СинП отвечает за соблюдение грамматической правильности предложений текста,

СемП отвечает за передачу свойства связности, которая при переводе реализуется, в частности, правильным выбором местоимений,

ИнфП отвечает за параметр осмысленности перевода в пределах ПО, релевантной для данного текста.

Наличие такой многоуровневой поддержки при анализе (понимании) и переводе позволяет разгрузить каждый отдельный уровень. Отдельное задание средств каждого уровня (грамматик, словарей, свойств правильных и допустимых структур и т. д.) допускает их независимую параллельную реализацию.

12.4.3. Словари. Словарный комплекс системы ФРАП содержит в настоящее время 10 словарей. При анализе каждый уровень вызывает свои словари. В построении СинП участвуют морфолого-синтаксический словарь, входами которого являются основы и словоформы, и словарь оборотов. СемП строится на основе двух семантических словарей (словаря термов и словаря отношений), входами в него являются словозначения. Переход к ИнфП связан с использованием тезаурусов, терминологических словников и предметно-ориентированных словарей.

Входные единицы словарей всех уровней могут быть не только простыми, но и сложными, состоящими из нескольких лексем (словари оборотов, терминов, словосочетаний и грамматических конструкций). Уровень сложной единицы диагностируется типом отношений между составляющими ее лексемами.

В описываемой версии системы словари по составу преимущественно общезыковые, но содержат и небольшие пласты специальной лексики (в основном относящиеся к области вычислительной техники, электроники, самолетостроения).

При включении компонента ПО и задании ПО-словарей вся «специальная» лексика может быть исключена из общих словарей.

Наиболее сложными по структуре и составу информации являются словари семантического уровня: французский семантический словарь (ФСС) термов и словарь отношений (СОТ). По сути дела они являются продолжением грамматико-семантического уровня.

Входом в словарную статью ФСС является лексическая единица (слово, оборот, фразеологическое сочетание, термин) в определенном значении. Значение слова (С) описывается в ФСС с разных точек зрения: грамматической, синтаксической, сочетаемостной и переводной. Это описание распределено более чем по 60 полям. Пять следующих полей являются обязательными для всех С: ЗГЛ (заглавное слово), ЗН (номер значения), СК (синтаксический класс), СХ (семантическая характеристика), РЭ (русский эквивалент).

Центральным понятием словарного описания в ФСС является семантическая валентность, или потенциальная смысловая связь С с другими словами, его семантическими актантами, которые могут находиться как в том же предложении, так и в других предложениях текста. Описание валентностей включает следующие сведения: 1) название смыслового отношения между С и его актантом (поле ВАЛ — валентность), например «содержание (А1, С)»; 2) семантические требования к кандидатам на заполнение валентности (поля СТ1—СТ3). Семантические требования формулируются в виде списка семантических признаков и смысловых отношений; 3) сильные синтаксические способы реализации валентностей (поле РВ). Синтаксический способ реализации валентности задается именем синтаксической связи и служебным словом (предлогом, союзом), оформляющим связь.

Кроме полей, непосредственно описывающих валентности С, имеются следующие группы полей, описывающие:

- индивидуальные ограничения на характеристики С и его актантов;
- смысловые отношения между актантами С;
- условия на выбор переводных эквивалентов;
- операции над СемП.

Инструментом интерпретации слабых связей является словарь отношений. Вход в него задается тремя параметрами: самим лексико-грамматическим элементом (предлог, союз, знак препинания), связывающим два узла, набором грамматических и семантических характеристик слабоуправляемого узла и набором таких же характеристик кандидата в управляющие; возможны уточняющие условия. На выходе — смысловое отношение и русские лексико-грамматические корреляты.

12.4.4. Грамматика и механизмы анализа. Синтаксический анализ в системе ФРАП основан на понятиях традиционной грамматики и предназначен для установления синтаксических функций слов и групп слов как членов предложения (подлежащее, дополнение, обстоятельство и др.). Формализация традиционного синтаксиса требует нетривиального сочетания формализмов зависимостей и составляющих для моделирования, например, таких объектов, как простые предложения в сложном, синтаксическая группа члена предложения. Синтаксический анализ реализуется как модульное процедурное описание, в котором управляющая процедура регламентирует порядок обработки фразы разделами грамматики. Модули, обрабатывающие определенные грамматические конструкции и явления, включаются в зависимости от ситуации во фразе. Сложность описания при этом распределяется между правилами процедурной и статической природы.

В фильтрах и некоторых других процедурах синтаксического анализа эффективно используются формальные свойства синтаксической структуры фразы — древность и проективность.

Перед синтаксическим анализом в системе ФРАП в общем случае не ставится задача получения дерева фразы любой ценой. Задача состоит в том, чтобы приблизиться к такой структуре настолько, насколько это позволяют средства синтаксического компонента и свойства конкретной фразы. Синтаксический раздел грамматики состоит из двух частей. Первой частью обрабатываются именные

словосочетания и осуществляется выделение простых предложений во фразе. Второй частью простые предложения обрабатываются независимо друг от друга в соответствии с типом предложения и его предполагаемым составом, отраженным в специально создаваемом для фразы метапредставлении каждого простого предложения, входящего во фразу. В случае, если во фразе не выделились однозначно простые предложения, она не может быть в полной мере обработана второй частью грамматики и должна быть переведена (в синтаксическом режиме) как последовательность именных словосочетаний и отдельных слов. По состоянию на 1987 г. таких фраз было не менее 20 %. Для их обработки должны быть введены дополнительные процедуры разрешения неоднозначности, но очевидно, что при этом сохранится определенный процент фраз, для полной обработки которых понадобятся нетривиальные процедуры обращения к семантике текста или к интерредактору.

В результате полностью проведенного синтаксического анализа строится связанное СинП в виде дерева зависимостей. Информация о слове как члене предложения выражена дважды: именем связи r в формуле вида $r(a, b)$, например ОПРЕДЕЛЕНИЕ (a, b), и именем синтаксической функции узла при самом слове. Это позволяет оперировать с информацией о синтаксической функции узла как при работе со структурой, так и при линейном анализе. Основное СинП (дерево зависимостей) дополняется специальным линейным метапредставлением фразы, которое формируется в ходе анализа и отражает линейные свойства единиц (состав узлов, перечень предложений, взаимное расположение слов и групп и т. д.). Алгоритмы анализа могут оперировать с этим представлением как со структурой составляющих.

Главной особенностью СинП системы ФРАП по сравнению с другими системами являются фиксации в явном виде всех видов обнаруженной и не снятой в ходе анализа неоднозначности. Различаются и фиксируются следующие пять типов неоднозначности: синтаксическая омонимия; грамматическая омонимия; структурная вариантность; составная вариантность 1 (связанная с неоднозначным членением сложных единиц типа «оборотов»); составная вариантность 2 (связанная с неоднозначностью выделения конца предложения).

Семантический анализ в системе ФРАП основан на представлении о семантических отношениях (СемО) как универсальном средстве фиксации смысла. В качестве языка описания смысла принят язык бинарных СемО, средства которого используются в описываемой версии системы не полностью.

Основным источником семантической информации является семантический словарь. Выделенные в нем значения образуют узлы семантического представления, а валентности этих значений — его связи.

В рамках принятого подхода понятия валентности, лексической функции рассматриваются не с синтаксической, а с семантической точки зрения как источник гипотез о СемО и форм их реализации на поверхностном уровне.

Семантический анализ (СемАн) в реализованной версии системы ограничен рамками предложения: он осуществляет интерпретацию СинП, обращаясь к различным полям словарной статьи семантического словаря и сравнивая их содержимое с единицами СинП.

В задачу построения СемП входит выбор для каждого слова того значения из всех имеющихся в словаре, которое реализовано в данном контексте, а также обнаружение смысловых отношений, стоящих за синтаксическими связями, установленными СинАн. Очевидно, что при решении этой задачи неоднозначность всех видов — постоянное и нормальное состояние.

Нетривиальность согласования словарной информации и синтаксического представления (с учетом всей неоднозначности) обусловила необходимость создания промежуточного синтактико-семантического представления (СинСемП) фразы, а также его форму. СинСемП строится как «наращивание» СинП путем создания семантического уровня описания его узлов, в который включается информация обо всех значениях слова в словаре и о валентных структурах этих значений: валентностях, способах реализации валентностей и кандидатах на их

заполнение. По завершении СемАн на базе СинСемП формируется СемП, которое содержит семантические узлы и эксплицированные связи между ними.

Процесс СемАн складывается из трех крупных этапов. На первом происходит достройка СинП до СинСемП с установлением всех возможных гипотез о заполнении валентностей. На втором этапе анализируются валентные структуры предикатов и рассматриваются различные виды неоднозначности. На третьем происходит интерпретация слабых связей по словарю отношений, являющемуся по сути грамматикой слабых связей.

Наиболее полное описание системы ФРАП см. в [Машинный, 1987].

12.5. О перспективах развития отечественных систем машинного перевода

Рассмотрение трех отечественных систем АП показывает, насколько велик разброс в способах решения известных переводческих трудностей; от способа их решения зависит общая конструкция системы. Проследим это различие по двум характерным для СМП аспектам: отношению к неоднозначности и представлению о следующем шаге.

Неоднозначность считается едва ли не главным бедствием АП, существует стремление как можно раньше диагностировать и подавлять ее. Система АМΠΑР здесь не исключение. Лексическая неоднозначность разрешается в ней рассмотрением локального контекста, предусмотренного в «схемах» разрешения неоднозначности (информация при лексеме). Вопрос о синтаксической неоднозначности возникает резко из-за отсутствия глобального синтаксического анализа; если она появилась, выбирается первое (достаточно случайное) решение.

Система ЭТАП справляется с неоднозначностью на уровне синтаксической структуры. В конце анализа может выдаваться несколько вариантов, но хорошая организация правил и алгоритмов работы с ними, а также стыковка с частной грамматической информацией, задаваемой в словаре, обеспечивают ранжирование вариантов: первым должен выдаваться лучший вариант; если он отбраковывается, берется следующий. Если вариант единственный, то он выходит на перевод.

В системе ФРАП неоднозначность — скорее конструктивный фактор: сам факт существования неоднозначности несет информацию, являясь особым структурным контекстом для рассмотрения других явлений. Такой контекст исчезнет, если будет построено несколько однозначных альтернативных вариантов. В полной модели неоднозначность и неопределенность любого промежуточного уровня должна сохраняться в течение всего анализа, так как неизвестно, по какому варианту пойдет развитие текста. Синтаксически однозначная структура может получить несколько вариантов интерпретации на семантическом уровне. Более того, единственный вариант синтаксического анализа, получивший хорошую интерпретацию на семантическом уровне локального масштаба, может быть отвергнут при семантическом целостностном анализе или на уровне построения информационного представления (ср. франц. *Choisir des yeux ouverts*, буквально: 'Выбирать открытые глаза' правильный перевод: «Голосовать с открытыми глазами»). Система выступает здесь как инструмент исследования свойств реальных текстов в аспекте неоднозначности.

Другое существенное различие между описанными системами — представление о следующем шаге развития системы. Для промышленной системы АМПАР — это, видимо, настройка на информационные потребности заказчиков, расширение словарей и тематик, улучшение эксплуатационных характеристик. В системе ЭТАП следующим логическим шагом должно быть построение глубинного синтаксического представления, достижение полноты и непротиворечивости лингвистической модели этого следующего уровня, что разработчики считают гарантией высокого качества АП. Для системы ФРАП следующий шаг — построение целостностного СемП на основе строящегося сейчас семантического пространства текста, освоение информационных механизмов, открывающих возможность не только лингвистических, сохраняющих эквивалентность преобразо-

ваний, но и информационных, меняющих объем поступающей на перевод информации. Задача перевода не решается только лингвистическими средствами (так, в систему ФРАП может быть включена другая синтаксическая модель, если она окажется более эффективной в каком-либо отношении). Семантическое представление целого текста должно обеспечить полноценный смысловой контекст для классического пофразного перевода, быть основой настройки системы на перевод текстов, принадлежащих разным предметным областям, быть основой для смыслового сжатия и последующего построения вторичных документов.

При хорошем обеспечении того большого и сложного эксперимента, которым является, по сути дела, любая работающая система машинного перевода, можно выработать наиболее эффективные комбинации методов и средств разработки новых систем.

Глава 13.

Особенности разработки и использования экспертных систем

Э. В. Попов

13.1. Назначение и принципы построения экспертных систем

В течение последнего десятилетия в рамках исследований по искусственному интеллекту (ИИ) сформировалось самостоятельное направление — *экспертные системы* (ЭС) или *инженерия знаний*. В задачу этого направления входят исследование и разработка программ (устройств), использующих знания и процедуры вывода для решения задач, являющихся трудными для людей-экспертов. В отличие от специализированных систем ИИ, которые были рассмотрены в предыдущих главах, ЭС могут быть отнесены к системам ИИ общего назначения (см. предисловие) — системам, которые не только исполняют заданные процедуры, но на основе метапроцедур поиска генерируют и используют процедуры решения новых конкретных задач.

Огромный интерес к ЭС со стороны пользователей вызван по крайней мере тремя причинами. Во-первых, они ориентированы на решение широкого круга задач в неформализованных областях, на приложения, которые до недавнего времени считались малодоступными для вычислительной техники. Во-вторых, с помощью ЭС специалисты, не знающие программирования, могут самостоятельно разрабатывать интересующие их приложения, что позволяет резко расширить сферу использования вычислительной техники. В-третьих, ЭС при решении практических задач достигают результатов, не уступающих, а иногда и превосходящих возможности людей-экспертов, не оснащенных ЭС.

В настоящее время ЭС применяются в различных областях человеческой деятельности. Наибольшее распространение ЭС получили в проектировании интегральных микросхем, в поиске неисправностей, в военных приложениях и автоматизации программирования. Применение ЭС позволяет: 1) при проектировании интегральных микросхем повысить (по данным фирмы NEC) производительность труда в 3—6 раз, при этом выполнение некоторых операций ускоряется в 10—15 раз; 2) ускорить поиск неисправностей в устройствах в 5—10 раз; 3) повысить производительность труда программистов (по данным фирмы Toshiba) в 5 раз; 4) при профессиональной подготовке сократить (без потери качества) в 8—12 раз затраты на индивидуальную работу с обучаемым.

В настоящее время ведутся разработки ЭС для следующих приложений: раннее предупреждение национальных и международных конфликтов и поиск компромиссных решений; принятие решений в кризисных ситуациях; охрана пра-

вопорядка; законодательство; образование; планирование и распределение ресурсов; системы организационного управления (кабинет министров, муниципалитет, учреждение) и т. д.

В 1985 г. в США было продано 12 000 ЭС, из них 11 000 простых ЭС и 1000 сложных [Walker et al., 1987]. При этом примерно 85 % затрат на приобретение ЭС приходится на сложные ЭС (см. § 13.2). В 1986 г. рынок ЭС в США составлял 130 млн. дол., в 1987 г. — 200 млн. дол., при этом мировой рынок ЭС на 1987 г. оценивается более чем в 400 млн. дол. [Бюллетень, 1988]. Рынок ЭС составлял и будет составлять по крайней мере до 1990 г. 90 % рынка всех программных продуктов ИИ [Мануэль, 1985а]. По прогнозам 1987 г. рынок ЭС в США составит в 1990 г. 1 млрд. дол., в 1995 г. 8—10 млрд. дол., а в 2000 г. 60 млрд. дол. В производстве ЭС и связанных с ними услугами в 1990 г. будет занято 100 000 чел. (в 1984 г. только 3000 чел.).

Исследования и разработки ЭС составляют основу программ по информатике развитых государств. На проведение этих программ выделены весьма большие средства. Например, на 10-летнюю стратегическую компьютерную программу (СКП) США запланировано выделить 1,6 млрд. дол., а на Esprit Европейского экономического сообщества (ЕЭС) — примерно 1,94 млрд. дол. По мере выполнения указанных программ интерес к ним возрастает. Так, в начале 1988 г. ЕЭС принял решение об удвоении ассигнований на Esprit, доведя их до 3,2 млрд. экю (примерно 3,9 млрд. дол.). В США ассигнования на СКП не изменились, однако (с учетом принятия новых программ и расходов частных фирм) затраты на ЭС только в 1987 г. составили более 1 млрд. дол. По прогнозу затраты США на ЭС в 1990 г. составят 4 млрд. дол. Следует подчеркнуть, что в США военные ведомства финансируют 70 % всех исследований по ЭС, промышленные фирмы — 15 % и кредитно-финансовые организации — 10 %.

Опрос 300 английских ведущих фирм в области информатики показал, что в 1985 г. 54 % фирм использовали ЭС, 20 % планировали их использование. В 1987 г. практически все компьютерные фирмы США имели лаборатории и отделения по ЭС. Крупнейшие фирмы США широко используют ЭС в своей деятельности. Так, в 1987 г. фирма IBM использовала 70 ЭС. В ближайшие годы число ЭС, используемых фирмой IBM, будет ежегодно удваиваться. Следует подчеркнуть, что фирма IBM обратила внимание на ИИ с существенным опозданием (только в 1984 г.). Однако уже в 1987 г. IBM стала мировым лидером в прикладном ИИ [За рубежом, 1988].

Несмотря на значительную стоимость разработки ЭС приносят существенный доход. Например, ежегодный доход от ЭС DEFT (проверка гигантских дисководов для ЭВМ фирмы IBM) составляет 12 млн. дол. (начальная стоимость системы 100 тыс. дол.), а от ЭС XCON (помогает продавцам систем VAX фирмы DEC определять конфигурацию системы) 25 млн. дол. (при стоимости разработки около 3 млн. дол.).

13.1.1. Назначение и особенности ЭС. Знания, которыми обладает специалист в какой-либо области (дисциплине), можно разделить на формализованные (точные) и неформализованные (неточные). *Формализованные знания* формулируются в книгах и руководствах в виде общих и строгих суждений (законов, формул, моделей, алгоритмов и т. п.), отражающих универсальные знания. *Неформализованные знания*, как правило, не попадают в книги и руководства в связи с их конкретностью, субъективностью и приблизительностью. Знания этого рода являются результатом обобщения многолетнего опыта работы и интуиции специалистов. Они обычно представляют собой многообразие эмпирических (эвристических) приемов и правил.

В зависимости от того, какие знания преобладают в той или иной области (дисциплине), ее относят к формализованным (если преобладают точные знания) или к неформализованным (если преобладают неточные знания) описательным областям. *Задачи*, решаемые на основе точных знаний, называют *формализованными*, а задачи, решаемые с помощью неточных знаний, — *неформализованными*. (Речь идет не о неформализуемых, а о неформализованных задачах, т. е.

о задачах, которые, возможно, и формализуемы, но эта формализация пока неизвестна.

Традиционное программирование в качестве основы для разработки программы использует алгоритм, т. е. формализованное знание. Поэтому до недавнего времени считалось, что ЭВМ не приспособлены для решения неформализованных задач. Расширение сферы использования ЭВМ показало, что неформализованные задачи составляют очень важный класс задач, вероятно, значительно больший, чем класс формализованных задач. Неумение решать неформализованные задачи сдерживает внедрение ЭВМ в описательные науки. По мнению академика Дородницына [Дородницын, 1985], основной задачей информатики является внедрение ее методов в описательные науки и дисциплины. На основании этого можно утверждать, что исследования в области ЭС занимают значительное место в информатике.

Ньюэлл [Newell, 1969] предложил относить к неформализованным задачам те, которые обладают одной или несколькими из следующих особенностей:

алгоритмическое решение задачи неизвестно (хотя, возможно, и существует) или не может быть использовано из-за ограниченности ресурсов ЭВМ (времени, памяти);

задача не может быть определена в числовой форме (требуется символьное представление);

цели задачи не могут быть выражены в терминах точно определенной целевой функции.

Как правило, неформализованные задачи обладают неполнотой, ошибочностью, неоднозначностью и (или) противоречивостью знаний (как данных, так и используемых правил преобразования).

Экспертные системы не отвергают и не заменяют традиционного подхода к программированию, они отличаются от традиционных программ тем, что ориентированы на решение неформализованных задач и обладают следующими особенностями:

алгоритм решений не известен заранее, а строится самой ЭС с помощью символических рассуждений, базирующихся на эвристических приемах;

ясность полученных решений, т. е. система «осознает» в терминах пользователя, как она получила решение;

способность анализа и объяснения своих действий и знаний;

способность приобретения новых знаний от пользователя-эксперта, не знающего программирования, и изменения в соответствии с ними своего поведения; обеспечение «дружественного», как правило, естественно-языкового (ЕЯ) интерфейса с пользователем.

Обычно к ЭС относят *системы, основанные на знаниях*, т. е. системы, вычислительная возможность которых является в первую очередь следствием их наращиваемой базы знаний (БЗ) и только во вторую очередь определяется используемыми методами [Watelman, 1986]. *Методы инженерии знаний* (методы ЭС) в значительной степени инвариантны тому, в каких областях они могут применяться. Области применения ЭС весьма разнообразны: военные приложения, медицина, электроника, вычислительная техника, геология, математика, космос, сельское хозяйство, управление, финансы, юриспруденция и т. д. Более критичны методы инженерии знаний к типу решаемых задач. В настоящее время ЭС используются при решении задач следующих типов: принятие решений в условиях неопределенности (неполноты), интерпретация символов и сигналов, предсказание, диагностика, конструирование, планирование, управление, контроль и др. (см. § 13.2).

13.1.2. Структура и режимы ЭС. Типичная ЭС состоит из следующих основных компонентов (рис. 13.1): решателя (интерпретатора), рабочей памяти (РП), называемой также базой данных (БД), базы знаний (БЗ), компонентов приобретения знаний, объяснительного и диалогового

База данных предназначена для хранения исходных и промежуточных данных решаемой в текущий момент задачи. Этот термин совпадает по названию, но не по смыслу с термином, используемым в информационно-поисковых систе-

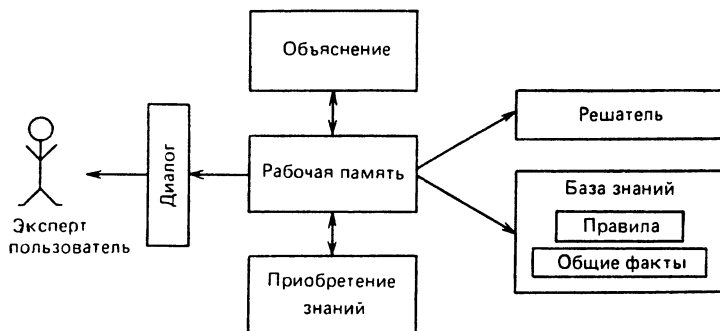


Рис. 13.1. Схема обобщенной экспертной системы

мах (ИПС) и системах управления базами данных (СУБД) для обозначения всех данных (и в первую очередь не текущих, а долгосрочных), хранимых в системе.

База знаний в ЭС предназначена для хранения долгосрочных данных, описывающих рассматриваемую область (а не текущих данных), и правил, описывающих целесообразные преобразования данных этой области.

Решатель, используя исходные данные из РП и знания из БЗ, формирует такую последовательность правил, которые, будучи примененными к исходным данным, приводят к решению задачи.

Компонент приобретения знаний автоматизирует процесс наполнения ЭС знаниями, осуществляемый пользователем-экспертом.

Объяснительный компонент объясняет, как система получила решение задачи (или почему она не получила решения) и какие знания она при этом использовала, что облегчает эксперту тестирование системы и повышает доверие пользователя к полученному результату.

Диалоговый компонент ориентирован на организацию дружелюбного общения со всеми категориями пользователей как в ходе решения задач, так и приобретения знаний, объяснения результатов работы.

В разработке ЭС участвуют представители следующих специальностей (см. § 13.4):

эксперт в той проблемной области, задачи которой будет решать ЭС;

инженер по знаниям — специалист по разработке ЭС;

программист — специалист по разработке инструментальных средств (ИС).

Необходимо отметить, что отсутствие среди участников разработки инженера по знаниям (т. е. его замена программистом) либо приводит к неудаче процесс создания ЭС, либо значительно удлиняет его.

Эксперт определяет знания (данные и правила), характеризующие проблемную область, обеспечивает полноту и правильность введенных в ЭС знаний.

Инженер по знаниям помогает эксперту выявить и структурировать знания, необходимые для работы ЭС, осуществляет выбор того ИС, которое наиболее подходит для данной проблемной области, и определяет способ представления знаний в этом ИС, выделяет и программирует (традиционными средствами) стандартные функции (типичные для данной проблемной области), которые будут использоваться в правилах, вводимых экспертом.

Программист разрабатывает ИС, содержащее в пределах все основные компоненты ЭС, осуществляет сопряжение ИС с той средой, в которой оно будет использовано.

Экспертная система работает в двух режимах: приобретения знаний и решения задач (называемом также режимом консультации или режимом использования ЭС).

В режиме приобретения знаний общение с ЭС осуществляет через посредничество инженера по знаниям эксперт. Эксперт описывает проблемную область в виде совокупности данных и правил. Данные определяют объекты, их характеристики и значения, существующие в области экспертизы. Правила определяют способы манипулирования данными, характерные для рассматриваемой проблемной области. Эксперт, используя компонент приобретения знаний, наполняет систему знаниями, которые позволяют ЭС в режиме решения самостоятельно (без эксперта) решать задачи из проблемной области.

Важную роль в режиме приобретения знаний играет объяснительный компонент. Именно благодаря ему эксперт на этапе тестирования (см. § 13.4) локализует причины неудачной работы ЭС, что позволяет эксперту целенаправленно модифицировать старые или вводить новые знания. Обычно объяснительный компонент сообщает следующее: как правила используют информацию пользователя; почему использовались или не использовались данные или правила; какие были сделаны выводы и т. п. Все объяснения делаются, как правило, на ограниченном естественном языке или языке графики.

Отметим, что режиму приобретения знаний при традиционном подходе к разработке программ соответствуют этапы алгоритмизации, программирования и отладки, выполняемые программистом. Таким образом, в отличие от традиционного подхода разработку программ осуществляет эксперт (с помощью ЭС), не владеющий программированием, а не программист.

В режиме консультации общение с ЭС осуществляет конечный показатель, которого интересует результат и (или) способ получения решения. Пользователь в зависимости от назначения ЭС может не быть специалистом в данной проблемной области, в этом случае он обращается к ЭС за советом, не умея получить ответ сам, или быть специалистом, в этом случае он обращается к ЭС, чтобы либо ускорить процесс получения результата, либо возложить на ЭС рутинную работу. Термин «пользователь» является многозначным, так как кроме конечного пользователя применять ЭС может и эксперт, и инженер по знаниям, и программист. Поэтому, когда хотят подчеркнуть, что речь идет о том, для кого делалась ЭС, используют термин «конечный пользователь».

В режиме консультации данные о задаче пользователя обрабатываются диалоговым компонентом, который выполняет следующие действия:

- распределяет роли участников (пользователя и ЭС) и организует их взаимодействие в процессе кооперативного решения задачи;

- преобразует данные пользователя о задаче, представленные на привычном для пользователя языке, во внутренний язык системы;

- преобразует сообщения системы, представленные на внутреннем языке, в сообщения на языке, привычном для пользователя (обычно это ограниченный естественный язык или язык графики).

После обработки данные поступают в РП. На основе входных данных из РП, общих данных о проблемной области и правил из БЗ решатель (интерпретатор) формирует решение задачи.

В отличие от традиционных программ ЭС в режиме решения задачи не только исполняет предписанную последовательность операций, но и предварительно формирует ее. Если ответ ЭС не понятен пользователю, то он может потребовать объяснения, как ответ получен.

13.2. Классификация экспертных систем

Экспертные системы как любой сложный объект можно определить только совокупностью характеристик. Выделим следующие характеристики ЭС: назначение; проблемная область; глубина анализа проблемной области; тип используемых методов и знаний; класс системы; стадия существования; инструментальные средства (ИС). Перечисленный набор характеристик не претендует на полноту (в связи с отсутствием общепринятой классификации), а определяет ЭС как целое, не выделяя отдельных компонентов (способ представления знаний, решение задачи и т. п.).

Назначение определяется следующей совокупностью параметров: цель создания ЭС — для обучения специалистов, для решения задач, для автоматизации рутинных работ, для тиражирования знаний экспертов и т. п.; основной пользователь — не специалист в области экспертизы, специалист, учащийся.

Проблемная область может быть определена совокупностью параметров: предметной областью и задачами, решаемыми в предметной области, каждый из которых может рассматриваться с точки зрения как конечного пользователя, так и разработчика ЭС.

С точки зрения пользователя, предметную область можно характеризовать описанием области в терминах пользователя, включающим наименование области, перечень и взаимоотношение подобластей и т. п., а задачи, решаемые существующими ЭС, — их типом. Обычно выделяют следующие *типы задач*:

интерпретация символов или сигналов — составление смыслового описания по входным данным;

предсказание — определение последствий наблюдаемых ситуаций;

диагностика — определение неисправностей (заболеваний) по симптомам;

конструирование — разработка объекта с заданными свойствами при соблюдении установленных ограничений;

планирование — определение последовательности действий, приводящих к желаемому состоянию объекта;

слежение — наблюдение за изменяющимся состоянием объекта и сравнение его показателей с установленными или желаемыми;

управление — воздействие на объект для достижения желаемого поведения.

С точки зрения разработчика целесообразно выделять *статические* и *динамические предметные области*. Предметная область называется статической, если описывающие ее исходные данные не изменяются во времени (точнее, рассматриваются как не изменяющиеся за время решения задачи). Статичность области означает неизменность описывающих ее исходных данных. При этом производные данные (выводимые из исходных) могут и появляться заново, и изменяться (не изменяя, однако, исходных данных). Если исходные данные, описывающие предметную область, изменяются за время решения задачи, то предметную область называют динамической. Кроме того, предметные области можно характеризовать следующими аспектами: числом и сложностью сущностей; их атрибутов и значений атрибутов; связностью сущностей и их атрибутов; полнотой знаний; точностью знаний (знания точны или правдоподобны; правдоподобность знаний представляется некоторым числом или высказыванием).

Решаемые задачи, с точки зрения разработчика ЭС, также можно разделить на статические и динамические. Будем говорить, что ЭС решает *динамическую* или *статическую* задачу, если процесс решения задачи изменяет или не изменяет исходные данные о текущем состоянии предметной области.

В подавляющем большинстве существующих ЭС исходят из предположения статичности предметной области и решают статические задачи, будем называть такие ЭС *статическими*. ЭС, которые имеют дело с динамическими предметными областями и решают статические или динамические задачи, будем называть *динамическими*. В настоящее время начинают появляться первые динамические ЭС. По нашему мнению, решение многих важнейших практических неформализованных задач возможно только с помощью динамических, а не статических ЭС. Следует подчеркнуть, что на традиционных (числовых) последовательных ЭВМ с помощью существующих методов инженерии знаний можно решать только статические задачи, а для решения динамических задач, составляющих большинство реальных приложений, необходимо использовать специализированные символьные ЭВМ (см. § 13.3).

Решаемые задачи, кроме того, можно характеризовать следующими аспектами: числом и сложностью правил, используемых в задаче; связностью правил; пространством поиска; количеством активных агентов, изменяющих предметную область; классом решаемых задач.

По степени сложности выделяют *простые* и *сложные правила*. К сложным относят правила, текст записи которых на естественном языке занимает 1/3 стра-

ницы и больше. Правила, текст которых занимает менее 1/3 страницы, относят к простым.

По степени связности правил задачи делят на связанные и малосвязные. К связным относятся задачи (подзадачи), которые не удается разбить на независимые задачи. Малосвязные задачи удается разбить на некоторое количество независимых подзадач.

Можно сказать, что степень сложности задачи определяется не просто общим количеством правил данной задачи, а количеством правил в ее наиболее связанной независимой подзадаче.

Пространство поиска может быть определено по крайней мере тремя аспектами: размером, глубиной и шириной. Размер пространства поиска дает обобщенную характеристику сложности задачи. Выделяют малые (до 10! состояний) и большие (свыше 10! состояний) пространства поиска. Глубина пространства поиска характеризуется средним числом последовательно применяемых правил, преобразующих исходные данные в конечный результат, ширина пространства — средним числом правил, пригодных к выполнению в текущем состоянии.

Количество активных агентов существенно влияет на выбор метода решения. Выделяют следующие значения данного аспекта: ни одного агента, один агент, несколько агентов.

Класс решаемых задач характеризует методы, используемые ЭС для решения задачи. Данный аспект в существующих ЭС принимает следующие значения: задачи расширения, доопределения, преобразования. Задачи расширения и доопределения являются статическими, а задачи преобразования — динамическими.

К задачам расширения относятся задачи, в процессе решения которых осуществляется только увеличение информации о предметной области, не приводящее ни к изменению ранее выведенных данных, ни к выбору другого состояния области. Типичной задачей этого класса являются задачи классификации.

К задачам доопределения относятся задачи с неполной или неточной информацией о реальной предметной области, цель решения которых — выбор из множества альтернативных текущих состояний предметной области того, которое адекватно исходным данным. В случае неточных данных альтернативные текущие состояния возникают как результат ненадежности данных и правил, что приводит к многообразию различных доступных выводов из одних и тех же исходных данных. В случае неполных данных альтернативные состояния являются результатом доопределения области, т. е. результатом предположений о возможных значениях недостающих данных.

К задачам преобразования относятся задачи, которые осуществляют изменения исходной или выведенной ранее информации о предметной области, являющиеся следствием изменений либо реального мира, либо его модели.

Большинство существующих ЭС решают задачи расширения, в которых нет ни изменений предметной области, ни активных агентов, преобразующих предметную область. Подобное ограничение неприемлемо при работе в динамических областях.

По степени сложности структуры ЭС делят на поверхностные и глубинные. *Поверхностные ЭС* представляют знания об области экспертизы в виде правил (условие → действие). Условие каждого правила определяет образец некоторой ситуации, при соблюдении которой правило может быть выполнено. Поиск решения состоит в выполнении тех правил, образцы которых сопоставляются с текущими данными (текущей ситуацией в РП). При этом предполагается, что в процессе поиска решения последовательность формируемых таким образом ситуаций не оборвется до получения решения, т. е. не возникнет неизвестной ситуации, которая не сопоставится ни с одним правилом. *Глубинные ЭС*, кроме возможностей поверхностных систем, обладают способностью при возникновении неизвестной ситуации определять с помощью некоторых общих принципов, справедливых для области экспертизы, какие действия следует выполнить.

По типу используемых методов и знаний ЭС делят на традиционные и гибкие. *Традиционные ЭС* используют в основном неформализованные методы

инженерии знаний и неформализованные знания, полученные от экспертов. *Гибридные ЭС* используют и методы инженерии знаний, и формализованные методы, а также данные традиционного программирования и математики.

Совокупность рассматриваемых выше характеристик позволяет определить особенности конкретной ЭС. Однако пользователи зачастую стремятся охарактеризовать ЭС каким-либо одним обобщенным параметром. В этой связи говорят о поколениях ЭС. В настоящее время выделяют ЭС первого и второго поколения. По нашему мнению, целесообразно говорить о трех поколениях ЭС. К *первому поколению* следует отнести статические поверхностные ЭС, ко *второму* — статические глубинные ЭС (иногда ко второму поколению относят гибридные ЭС), а к *третьему* — динамические ЭС (вероятно, они, как правило, будут глубинными и гибридными).

В последнее время выделяют два больших класса ЭС (существенно отличающихся по технологии их проектирования), которые мы условно называем простыми и сложными ЭС. *Простая ЭС* может быть охарактеризована следующими значениями основных параметров: поверхностная ЭС; традиционная ЭС (реже гибридная); выполненная на персональной ЭВМ (ПЭВМ); коммерческая стоимостью от 100 до 25 тыс. дол.; стоимость разработки от 50 тыс. до 300 тыс. дол.; время разработки от 3 мес. до одного года (при использовании развитых ИС); от 200 до 1000 правил. *Сложная ЭС* может быть охарактеризована следующими значениями параметров: глубинная ЭС; гибридная ЭС; выполненная либо на символьной ЭВМ, либо на мощной универсальной ЭВМ, либо на интеллектуальной рабочей станции; коммерческая стоимостью от 50 тыс. до 1 млн. дол.; средняя стоимость разработки 5—10 млн. дол.; время разработки от 1 до 5 лет; от 1500 до 10 тыс. правил.

Стадия существования характеризует степень проработанности и отлаженности ЭС. Обычно выделяют следующие стадии: демонстрационный прототип; исследовательский прототип; действующий прототип; промышленная система; коммерческая система.

Демонстрационным прототипом называют ЭС, которая решает часть требуемых задач, демонстрируя жизнеспособность метода инженерии знаний. При наличии развитых ИС для разработки демонстрационного прототипа требуется в среднем примерно 1—2 мес., а при отсутствии — 12—18 мес. Демонстрационный прототип работает, имея в БЗ 50—100 правил. Развитие демонстрационного прототипа приводит к исследовательскому прототипу.

Исследовательским прототипом называют систему, которая решает все требуемые задачи, но неустойчива в работе и не полностью проверена. На доведение системы до стадии исследовательского прототипа уходит 3—6 мес. Исследовательский прототип обычно имеет в БЗ 200—500 правил, описывающих проблемную область.

Действующий прототип надежно решает все задачи, но для решения сложных задач может потребоваться чрезмерно много времени и (или) памяти. Для доведения системы до стадии действующего прототипа требуется 6—12 мес., при этом количество правил в БЗ увеличивается до 500—1000.

Экспертная система, достигшая *промышленной стадии*, обеспечивает высокое качество решений всех задач при минимуме времени и памяти. Обычно процесс преобразования действующего прототипа в промышленную систему состоит в расширении БЗ (до 1000—1500 правил) и переписывании программ с использованием более эффективных ИС, например в перепрограммировании на языках низкого уровня. Для доведения ЭС от начала разработки до стадии промышленной системы требуется 1—1,5 года.

Обобщение задач, решаемых ЭС на стадии промышленной системы, позволяет перейти к стадии *коммерческой системы* — к системе, пригодной не только для собственного использования, но и для продажи различным потребителям. Для доведения системы до коммерческой стадии требуется 1,5—3 года и 0,3—5 млн. дол. При этом в БЗ системы 1500—3000 правил.

Инструментальные средства определяют *программные* и *аппаратные* средства, используемые в рассматриваемой ЭС (см. § 13.3).

13.3. Классификация инструментальных средств

На проектирование и создание одной ЭС ранее требовалось 20—30 чел.-лет. В настоящее время имеется ряд средств, ускоряющих создание. Эти средства называют инструментальными (ИС) или инструментарием. Использование ИС сокращает время разработки ЭС в 3—5 раз.

В широком толковании в инструментарий включают и аппаратуру, ориентированную на разработку ЭС (аппаратный инструментарий). Экспертные системы выполняют на ЭВМ следующие типов: общего назначения; ПЭВМ; интеллектуальные рабочие станции (т. е. рабочие станции типа Sun, Apollo и др., снабженные эффективными ИС для создания ЭС); последовательные символьные ЭВМ типа ЛИСП-машин (Symbolic-3670, Alpha, Explorer, Xerox 1100 и др.) и Пролог-машин; параллельные символьные ЭВМ (Connection, Dado, Faim, Nureg Cube и др.).

Программные ИС будем определять следующей совокупностью характеристик: назначение; стадия существования; тип ИС; тип используемых методов и знаний; универсальность; основные свойства; среда функционирования.

Назначение определяет, для работы в каких проблемных областях и для создания какой стадии ЭС (см. § 13.2) предназначено ИС.

По степени отработанности ИС обычно выделяют три стадии существования: экспериментальная, исследовательская; коммерческая. *Экспериментальные ИС* создаются для решения узких специфических задач и редко проверяются на других задачах, обычно они работают медленно и неэффективно. Следующей стадией является *исследовательская*. Средства, достигшие этой стадии, обычно тщательно проверены, имеют документацию и поддерживаются разработчиком, однако они еще могут действовать медленно и неэффективно. Исследовательские ИС используются при разработке прототипов ЭС. Высшей стадией существования ИС является *коммерческая*. Эта стадия достигают те ИС, которые всесторонне и тщательно проверены, хорошо документированы, сопровождаются разработчиком, являются быстрыми и обладают удобным интерфейсом с пользователем.

По *типу* ИС классифицируют следующим образом [Waterman, 1986]: 1) символьные языки программирования, ориентированные на создание ЭС и систем ИИ (например, LISP, INTERLISP, SMALLTALK); 2) языки инженерии знаний, т. е. языки высокого уровня, ориентированные на построение ЭС (например, OPS-5, LOOPS, Пролог, KES); 3) системы, автоматизирующие разработку (проектирование) ЭС (например, KEE, ART, TEIRESIAS, AGE, TIMM), их часто называют окружением (environment) для разработки систем ИИ, ориентированных на знания; 4) оболочки ЭС (или пустые ЭС) — ЭС, не содержащие знаний ни о какой проблемной области (например, ЭКСПЕРТИЗА, ЕМУСIN, ЭКО, ЭКСПЕРТ).

В приведенной классификации ИС перечислены в порядке убывания затрат, необходимых на создание с их помощью конкретной ЭС. Действительно, при использовании ИС первого типа в задачу разработчика входит программирование всех компонентов ЭС на языке довольно низкого уровня. Использование ИС второго типа позволяет значительно повысить уровень языка, что, как правило, приводит к некоторому снижению эффективности. Инструментальные средства третьего типа позволяют разработчику не программировать все или часть компонентов ЭС, а выбрать их из заранее составленного набора. При применении ИС четвертого типа разработчик ЭС полностью освобождается от работ по созданию программ, так как берет готовую пустую ЭС.

При использовании ИС третьего и особенно четвертого типа могут возникнуть следующие проблемы: управляющие стратегии, вложенные в процедуры вывода ИС, могут не соответствовать методам решения, которые использует эксперт, взаимодействующий с данной ЭС, что может приводить к неэффективному, а возможно, и неправильным решениям; язык представления знаний, принятый в ИС, может не подходить для данного приложения.

Подчеркнем, что развитие систем, автоматизирующих разработку ЭС, приводит к появлению ИС, которые можно назвать *настраиваемыми оболочками*. Эти ИС позволяют разработчику использовать оболочку не как нечто неизменное (как имело место ранее в EMYCIN, KAS и др.), а генерировать оболочку из множества механизмов, имеющихся в ИС. Типичными примерами таких ИС являются КЕЕ, АРТ, ЭКСПЕРТИЗА, ГЛОБ.

Инструментальные средства можно классифицировать и по классам ЭС на: ИС для создания простых ЭС; ИС для создания сложных ЭС. В настоящее время, как правило, ИС первого типа разрабатываются на ПЭВМ, а второго — на символьных ЭВМ, ЭВМ общего назначения и интеллектуальных рабочих станциях.

По типу используемых методов и знаний ИС делятся, так же как и ЭС, на *традиционные* — использующие только методы и способы представления инженерии знаний, и *гибридные* — сочетающие подходы инженерии знаний с подходами, развитыми в традиционном программировании при представлении данных и использовании подпрограмм.

Универсальность задается совокупностью двух параметров: универсальностью представления знаний и универсальностью функционирования. Универсальность представления характеризует способ (модель) представления знаний в ИС, принимает следующие значения: единое представление — ИС использует одну модель; интегральное представление — ИС допускает интегральное использование нескольких моделей; универсальное представление — ИС допускает интегральное использование всех основных моделей представления. К основным моделям представления относятся: правила (продукции); фреймы или семантические сети; логические модели (исчисление предикатов). Примерами ИС, в которых используется единое представление, является Пролог, интегральное представление — CENTAUR, а универсальное — КЕЕ, АРТ.

Универсальность функционирования характеризует механизмы (парадигмы), определяющие, как в ИС задается поведение (функционирование) системы, принимает следующие значения: единый механизм функционирования — ИС использует один механизм функционирования; интегральное функционирование — ИС допускает интегральное использование нескольких механизмов функционирования; универсальное функционирование — ИС допускает интегральное использование всех основных механизмов. К основным механизмам нескольких (парадигмам программирования) относятся: процедурное программирование; объектное программирование; программирование, ориентированное на данные; программирование, ориентированное на правила. Примерами ИС, в которых реализован единый механизм функционирования, являются LISP, EMYCIN, интегральное функционирование OPS-83, универсальный механизм LOOPS. В связи с тем, что парадигмы программирования мало известны, поясним их.

Процедурное программирование наиболее широко распространено среди существующих сегодня языков программирования (см., например, Си). Здесь явно выделяют два вида различных сущностей: процедуры, являющиеся тем, что задает поведение (функционирование) программы и выполняющие активную роль; данные, являющиеся тем, что обрабатывается способом, предписанным процедурами, и выполняющие пассивную роль. Способность составлять процедуры из команд (операторов) и вызывать их — ключ функционирования, использующего данную парадигму. Особенностью ее являются боковые эффекты, возникающие в тех случаях, когда различные процедуры, использующие общие данные, независимо их изменяют.

Парадигма объектного программирования в отличие от процедурной не разделяет программу на процедуры и данные. Здесь (см., например, SMALL-TALK [Goldberg et al., 1982]) программа организуется вокруг сущностей (объектов), которые включают локальные процедуры (методы) и локальные данные (переменные). Поведение (функционирование) в этой парадигме организуется путем пересылки сообщений между объектами. Объект, получив сообщение, осуществляет его локальную интерпретацию, основываясь на локальных процедурах и данных. Объектная парадигма удобна в тех приложениях, где

взаимодействие сущностей сведено к унифицированным протоколам. Важным свойством языков, использующих данную парадигму, является сеть наследований, которая существенно упрощает определение новых объектов, почти подобных известным.

В обеих рассмотренных парадигмах активная роль в организации поведения отводится процедурам, а не данным. Причем процедура активизируется либо ее вызовом, либо посылкой ей сообщения. Подобные способы задания поведения удобны для описаний детерминированной последовательности действий либо одного процесса, либо нескольких, но строго взаимосвязанных процессов.

В *программировании, ориентированном на данные*, активную роль играют данные, а не процедуры. Здесь со структурами активных данных связывают некоторые процедуры (действия), которые активизируются тогда, когда осуществляется доступ к этим данным. Описанный механизм в некотором смысле обратный механизму посылки сообщений. Действительно, сообщение является явным способом указать объекту, какие действия следует выполнять (эти действия в качестве бокового эффекта могут изменить значения некоторых данных (переменных)), а отнесение данных к активным является явным способом указания на контроль за доступом к ним, при этом в качестве бокового эффекта может быть послано сообщение процедуре. Программирование, ориентированное на данные, позволяет организовать поведение малозависимых процессов, что трудно реализовать в процедурной и объектной парадигмах. Малая зависимость процессов означает, что они могут рассматриваться и программироваться отдельно. Однако при использовании парадигмы, управляемой данными, эти независимо запрограммированные процессы могут взаимодействовать между собой без их изменения, т. е. без перепрограммирования.

В *программировании, ориентированном на правила*, поведение определяется множеством правил вида условие — действие. Условие задает образ данных, при возникновении которого действие правила может быть выполнено. Правила в данной парадигме играют такую же роль, как и подпрограммы в процедурной парадигме. Однако если в процедурной парадигме поведение задается детерминированной последовательностью процедур, не зависящей (для широкого класса обрабатываемых данных) от значений этих данных, то в парадигме, ориентированной на правила, поведение (последовательность действий) задается не заранее предписанной последовательностью правил, а формируется на основе тех данных и их значений, которые в текущий момент обрабатываются программой (системой). Формирование поведения осуществляется по следующей схеме. Условия правил сопоставляются с текущими данными, и те правила, условия которых удовлетворяются значениями текущих данных, становятся претендентами на выполнение. Затем по определенному критерию осуществляется выбор одного правила среди претендентов и выполнение его (т. е. выполнение действия, указанного в правой части правила). Подчеркнем, что если система (устройство) допускает параллельную обработку, то все правила-претенденты могут выполняться одновременно. Таким образом, правила выполняются в порядке, предписываемом теми образцами данных, которые, как правило, мало-зависимы друг от друга. Итак, подход, ориентированный на правила, удобен для описания поведения, гибко и разнообразно реагирующего на большое многообразие состояний данных.

Основные свойства определяют особенности ИС с точки зрения реализации компонентов ЭС.

Среда функционирования определяет тип ЭВМ, на которой реализовано ИС, тип операционной системы, в среде которой ИС работает, и используемый язык программирования.

13.4. Методология разработки экспертных систем

Разработка (проектирование) ЭС существенно отличается от разработки обычного программного продукта. Опыт разработки ранних ЭС показал, что использование при их разработке методологии, принятой в традиционном про-

граммировании, либо чрезмерно затягивает процесс создания ЭС, либо вообще приводит к отрицательному результату. Дело в том, что неформализованность задач, решаемых ЭС, отсутствие завершенной теории ЭС и методологии их разработки приводят к необходимости модифицировать принципы и способы построения ЭС в ходе процесса разработки по мере того, как увеличивается знание разработчиков о проблемной области.

Перед тем как приступить к разработке ЭС, инженер по знаниям должен рассмотреть вопрос, следует ли разрабатывать ЭС для данного приложения. В обобщенном виде ответ может быть таким: использовать ЭС следует тогда, когда разработка ЭС возможна, оправдана и методы инженерии знаний соответствуют решаемой задаче. Ниже будут уточнены использованные понятия «возможно», «оправдано», «соответствует».

Чтобы разработка ЭС была возможной (для данного приложения), необходимо одновременное выполнение по крайней мере следующих требований: существуют эксперты в данной области, которые решают задачу значительно лучше, чем начинающие специалисты;

эксперты должны сходиться в оценке предлагаемого решения, иначе нельзя будет оценить качество разработанной ЭС;

эксперты должны уметь выразить на естественном языке (вербализовать) и объяснить используемые ими методы, в противном случае трудно рассчитывать на то, что знания экспертов будут «извлечены» и вложены в ЭС;

задача, возложенная на ЭС, требует только рассуждений, а не действий (если требуются действия, то необходимо объединять ЭС с роботами);

задача не должна быть слишком трудной, ее решение должно занимать у эксперта несколько часов, а не дней или недель;

задача, хотя и не должна быть выражена в формальном виде, но все же должна относиться к достаточно «понятной» и структурированной области, т. е. должны быть выделены основные понятия, отношения и известные (хотя бы эксперту) способы получения решения задачи;

решение задачи не должно в значительной степени использовать «здравый смысл» (т. е. широкий спектр общих сведений о мире и о способе его функционирования, которые знает и умеет использовать любой нормальный человек), так как подобные знания пока не удастся (в достаточном количестве) вложить в системы ИИ.

Использование ЭС в данном приложении может быть возможно, но не оправдано. Применение ЭС может быть оправдано одним из следующих факторов:

решение задачи принесет значительный эффект, например использование ЭС для поиска полезных ископаемых в случае успеха может принести доход в сотни миллионов рублей;

использование человека-эксперта невозможно либо из-за недостаточного количества экспертов, либо из-за необходимости выполнять экспертизу одновременно в различных местах;

при передаче информации к эксперту происходит недопустимая потеря времени или информации;

при необходимости решать задачу в окружении, враждебном для человека.

Приложение соответствует методам ЭС, если решаемая задача обладает совокупностью следующих характеристик:

может быть естественным образом решена посредством манипуляции с символами (т. е. с помощью символических рассуждений), а не с числами (как принято в математических методах и в традиционных программах);

должна иметь эвристическую (не алгоритмическую) природу, т. е. ее решение должно сводиться к применению эвристических правил; задачи, которые могут быть гарантированно решены (с соблюдением заданных ограничений) с помощью некоторых формальных процедур, не подходят для применения ЭС;

должна быть достаточно сложной, чтобы оправдать затраты на разработку ЭС, однако не должна быть чрезмерно сложной (решение занимает у эксперта часы, а не недели), чтобы ЭС могла ее решить;

должна быть достаточно узкой, чтобы решаться методами инженерии знаний, и практически значимой.

При разработке ЭС используется концепция «быстрого прототипа». Суть ее состоит в том, что разработчики не пытаются сразу создать конечный продукт. На начальном этапе они создают прототип (прототипы) ЭС, который должен удовлетворять двум противоречивым требованиям: с одной стороны, решать типичные задачи конкретного приложения, а с другой — время и трудоемкость его разработки должны быть весьма незначительными, чтобы можно было максимально запараллелить процесс накопления и отладки знаний (осуществляемый экспертом) с процессом выбора (разработки) программных средств (осуществляемым инженером по знаниям и программистом). Для удовлетворения указанных требований при создании прототипа, как правило, используются разнообразные ИС, ускоряющие процесс проектирования.

Прототип должен продемонстрировать пригодность методов инженерии знаний для данного приложения. В случае успеха эксперт с помощью инженера по знаниям расширяет знания прототипа о проблемной области. При неудаче может потребоваться разработка нового прототипа или разработчики могут прийти к выводу о непригодности методов инженерии знаний для данного приложения. По мере увеличения знаний прототип может достичь такого состояния, когда он успешно решает все задачи данного приложения. Преобразование прототипа ЭС в конечный продукт обычно приводит к перепрограммированию ЭС на языках низкого уровня, обеспечивающих как повышение быстродействия ЭС, так и уменьшение требуемой памяти. Трудоемкость и время создания ЭС в значительной степени зависят от типа используемого ИС.

В ходе работ по созданию ЭС сложилась определенная технология их разработки, включающая шесть следующих этапов [Waterman, 1986; Попов, 1987]: идентификация, концептуализация, формализация, выполнение, тестирование, опытная эксплуатация (рис. 13.2).

На этапе идентификации определяются задачи, которые подлежат решению, выявляются цели разработки, ресурсы, эксперты и категории пользователей.

На этапе концептуализации проводится содержательный анализ проблемной области, выявляются используемые понятия и их взаимосвязи, определяют-ся методы решения задач.

На этапе формализации определяются способы представления всех видов знаний, формализуются основные понятия, определяются способы интерпретации знаний, моделируется работа системы, оценивается адекватность целям системы зафиксированных понятий, методов решения, средств представления и манипулирования знаниями.

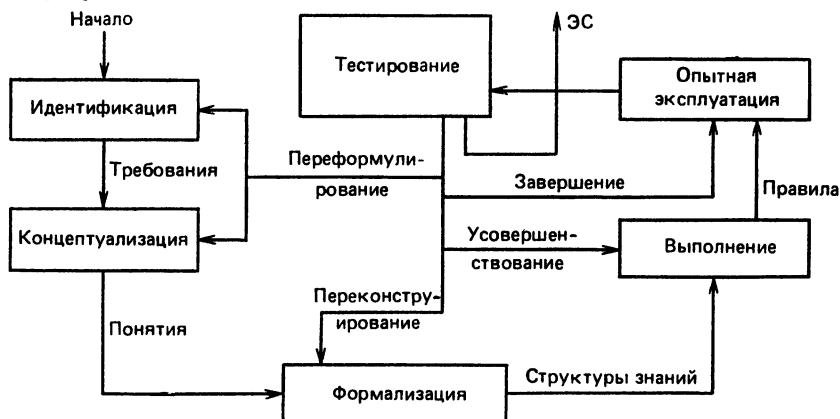


Рис. 13.2. Технология разработки ЭС

На этапе выполнения осуществляется наполнение экспертом БЗ системы. Процесс приобретения знаний разделяют на извлечение знаний из эксперта, организацию знаний, обеспечивающую эффективную работу системы, и представление знаний в виде, понятном ЭС. Эвристический характер знаний приводит к тому, что процесс их приобретения является весьма трудоемким.

На этапе тестирования эксперт (и инженер по знаниям) в интерактивном режиме, используя диалоговые и объяснительные средства, проверяет компетентность ЭС. Процесс тестирования продолжается до тех пор, пока эксперт не решит, что система достигла требуемого уровня компетентности.

На этапе опытной эксплуатации проверяется пригодность ЭС для конечных пользователей. По результатам этого этапа может потребоваться существенная модификация ЭС.

Процесс создания ЭС не сводится к строгой последовательности перечисленных выше этапов. В ходе разработки приходится неоднократно возвращаться на более ранние этапы и пересматривать принятые там решения (см. § 13.5).

13.5. Этапы разработки экспертных систем

Рассмотрим более детально этапы разработки ЭС (рис. 13.2).

13.5.1. Этап идентификации. На данном этапе идентифицируется задача (задачи), определяются участники процесса проектирования и их роли, ресурсы, цели.

Определение участников и их ролей сводится к определению количества экспертов и инженеров по знаниям, а также форм их взаимоотношения. Обычно в основном цикле разработки ЭС участвует не менее трех-четырех человек — один эксперт, один или два инженера по знаниям и один программист, привлекаемый для модификации и согласования инструментальных средств. К процессу разработки ЭС могут привлекаться и другие участники. Например, инженер по знаниям может привлекать других экспертов, чтобы убедиться в правильности своего понимания основного эксперта, представительности тестов, демонстрирующих особенности рассматриваемой задачи, совпадении взглядов различных экспертов на качество предлагаемых решений. Кроме того, для сложных приложений целесообразно привлекать к основному циклу разработки несколько экспертов. Однако в этом случае, как правило, требуется, чтобы один из экспертов отвечал за непротиворечивость знаний, сообщаемых всеми экспертами, т. е. выполнял функции руководителя экспертов.

Что касается формы взаимоотношения экспертов и инженеров, то применяются следующие формы: эксперт выполняет роль информирующего или эксперт выполняет роль учителя, а инженер — ученика. По нашему мнению, форма учитель-ученик больше соответствует методологии ЭС. Вне зависимости от выбранной формы взаимоотношений инженер по знаниям должен быть готов и способен изучить особенности той проблемной области, в рамках которой предстоит работать создаваемой ЭС. Несмотря на то, что основу знаний ЭС будут составлять знания эксперта, для достижения успеха инженер по знаниям должен использовать (особенно в начале работы) дополнительные источники знаний в виде книг, инструкций, отчетов, которые ему рекомендовал эксперт.

Советы инженеру.

1. До начала интенсивного взаимодействия с экспертом овладейте основами задачи, для решения которой разрабатывается ЭС. Несоблюдение данного совета приведет к чрезмерному затягиванию этапа идентификации задачи (см. § 13.7) и к тому, что эксперт будет терять веру в конечный успех.

2. Помните, что не любой специалист, работающий в интересующей вас области, является экспертом (необходимо убедиться в квалификации претендента на роль эксперта).

3. Не любой, являющийся экспертом, подходит для создания ЭС, от эксперта требуется по крайней мере следующее: доступность — эксперт должен быть готов к длительному общению с инженером (от полугода до года, по 3 раза в неделю, по 2—3 часа на каждое общение); вера в успех — эксперт должен осознавать сложность задачи, верить в ее важность и не терять надежды на успех (поддержание веры в успех входит в задачу инженера по знаниям).

Идентификация задачи заключается в составлении неформального (вербального) описания решаемой задачи. В этом описании указываются: общие характеристики задачи; подзадачи, выделяемые внутри данной задачи; ключевые понятия (объекты), характеристики и отношения; входные (выходные) данные; предположительный вид решения; знания, релевантные решаемой задаче; примеры (тесты) решения задачи.

Цель этапа идентификации задачи — характеризовать задачу и структуру поддерживающих ее знаний и таким образом обеспечить начальный импульс для развития БЗ. Если исходная задача оказывается слишком сложной с точки зрения имеющихся ресурсов, то этап идентификации может потребовать нескольких итераций.

В ходе идентификации задачи (задач) необходимо ответить на следующие вопросы: какие задачи предлагается решать ЭС и как они могут быть охарактеризованы и определены; на какие подзадачи разбивается каждая задача и какие данные они используют; каковы основные понятия и взаимоотношения, используемые при формулировании и решении задачи; какие знания релевантны решаемой задаче; какие ситуации препятствуют решению; как эти препятствия будут влиять на ЭС.

В процессе идентификации задачи инженер и эксперт работают в тесном контакте. Начальное неформальное описание задачи экспертом используется инженером по знаниям для уточнения терминов и ключевых понятий. Эксперт уточняет описание задачи, объясняет, как решать ее и какие рассуждения лежат в основе решения. После нескольких циклов уточняющих описаний эксперт и инженер по знаниям получают окончательное неформальное описание задачи.

При проектировании ЭС типичными ресурсами являются: источники знаний, время разработки, вычислительные средства (возможности ЭВМ и программного ИС) и объем финансирования. Для достижения успеха эксперт и инженер должны использовать при построении ЭС все доступные им источники знаний. Для эксперта источниками знаний могут быть его предшествующий опыт по решению задачи, книги, конкретные примеры задач и использованных решений, а для инженера по знаниям — опыт в решении аналогичных задач, методы решения и представления знаний, программное ИС. При определении (назначении) времени разработки необходимо иметь в виду, что сроки разработки и внедрения ЭС составляют (за редким исключением) не менее года (при трудоемкости 5 чел.-лет). Задача определения ресурсов является весьма важной, поскольку ограниченность какого-либо ресурса существенно влияет на процесс разработки. Так, при недостаточном объеме финансирования предпочтение может быть отдано не разработке оригинальной новой системы, а адаптации существующей.

Идентификация целей заключается в формулировании в явном виде целей построения ЭС. При этом важно отличать цели, ради которых создается ЭС, от задач, которые она должна решать. Примерами возможных целей являются: формализация неформальных знаний эксперта; улучшение качества решений, принимаемых экспертом; автоматизация рутинных аспектов работы эксперта (пользователя); тиражирование знаний эксперта.

На первом этапе инженер по знаниям должен ответить на основной вопрос: подходят методы инженерии знаний для решения предложенной задачи? Для положительного ответа на этот вопрос необходимо, чтобы задача относилась к узкой, специальной области знаний и не требовалось для ее решения использовать то, что принято называть «здравым смыслом». Кроме того, качество ЭС

зависит в конечном счете от уровня сложности решаемой задачи и ясности ее формулировки. Задача не должна быть ни слишком легкой, ни слишком трудной. Говоря другими словами, назначение ЭС в том, чтобы решать некоторую задачу из данной области, а не в том, чтобы быть экспертом в этой области. Для обеспечения ясности формулировки задачи следует обратить внимание на точное описание входа-выхода и на наличие разнообразных примеров решений рассматриваемой задачи.

13.5.2. Этап концептуализации. На этом этапе эксперт и инженер по знаниям эксплицируют ключевые понятия, отношения (упомянутые на этапе идентификации) и характеристики, необходимые для описания процесса решения задачи. На этом этапе определяются следующие особенности задачи: типы доступных данных; исходные и выводимые данные, подзадачи общей задачи; используемые стратегии и гипотезы; виды взаимосвязей между объектами проблемной области, типы используемых отношений (иерархия, причина-следствие, часть-целое и т. п.); процессы, используемые в ходе решения задачи; типы ограничений, накладываемых на процессы, используемые в ходе решения; состав знаний, используемых для решения задачи; состав знаний, используемых для оправдания (объяснения) решения.

Совет инженеру.

Для определения перечисленных характеристик задачи целесообразно составить детальный протокол действий и рассуждений эксперта в процессе решения хотя бы одной конкретной задачи.

Протокол обеспечивает инженера по знаниям словарем терминов (объектов) и некоторым приблизительным представлением о тех стратегиях, которые использует эксперт, кроме того, помогает ответить на многие другие вопросы, возникающие в ходе разработки. На этом этапе инженер по знаниям рассматривает вопросы, относящиеся к представлению знаний и методам решения, но говорить о выборе конкретных способов и методов здесь еще рано.

На этапе концептуализации (как и на этапе идентификации) может потребоваться несколько повторных взаимодействий между экспертом и инженером по знаниям, что приводит к значительным затратам времени. Опыт разработок ЭС убеждает в том, что на данном этапе невозможно, да и не нужно добиваться корректности и полноты разрабатываемой ЭС. Необходимо как можно раньше переходить к следующим этапам (формализации и выполнению). На данном этапе у инженера по знаниям появляются некоторые соображения о представлении знаний и ИС.

Совет инженеру.

Не следует окончательный выбор представления и инструментария делать на этапе концептуализации. Тем не менее уже на этом этапе инженер может попытаться применить имеющиеся у него идеи для создания прототипа некоторой подзадачи.

Инженер по знаниям на этапе концептуализации должен обеспечить достаточность введенных ключевых понятий и отношений для описания всех имеющихся примеров решения рассматриваемой проблемы.

13.5.3. Этап формализации. На данном этапе все ключевые понятия и отношения, введенные на этапе концептуализации, выражаются на некотором формальном языке, предложенном (выбранном) инженером по знаниям. Здесь он определяет, подходят ли имеющиеся ИС для решения рассматриваемой проблемы или необходимы оригинальные разработки. Выходом этапа формализации является описание того, как рассматриваемая задача может быть представлена в выбранном или разработанном формализме.

Процесс формализации зависит от трех основных факторов: структуры пространства поиска, характеризующей особенности решаемой задачи; модели, лежащей в основе процесса решения задачи; свойств данных рассматриваемой задачи.

Чтобы понять структуру пространства поиска, необходимо формализовать понятия (объекты, их характеристики и значения) и определить, как они могут связываться друг с другом при образовании гипотез, используемых для на-

правления поиска. При этом необходимо ответить на следующие вопросы: являются понятия примитивными или имеют внутреннюю структуру; необходимо ли представлять причинные и пространственно-временные отношения между понятиями и должны ли они быть представлены явно; необходима ли иерархия гипотез; относится коэффициент определенности (или другие средства для выражения мнения) только к окончательным гипотезам или он необходим для промежуточных гипотез; необходимо ли рассматривать понятия и процессы на различных уровнях абстракции?

Важный шаг в процессе формализации знаний — построение модели исследуемой задачи, так как наличие модели позволяет генерировать решение. Могут быть использованы как поведенческие, так и математические модели. Если эксперт использует при рассуждении или обосновании решения хотя бы простейшую поведенческую модель, то это позволяет выработать важные понятия и отношения. Если же эксперт использует математическую модель (аналитическую или статистическую), то она может быть непосредственно включена в ЭС как для формирования решения, так и для объяснения (оправдания) причинных отношений, обнаруживаемых в БЗ.

Для формализации знаний весьма важно понимать природу данных проблемной области. Необходимо определить свойства данных, которые существенно влияют на решение исходной проблемы. Это могут быть следующие свойства: данные объясняются или нет в терминах гипотез; тип отношений между данными (причинные, определительные, корреляционные); знание типа отношения помогает объяснить, как взаимосвязаны данные, гипотезы и цели в процессе решения; данные редки (обильны) или недостаточны (избыточны); данные определены или нет, т. е. требуется ли использование коэффициентов определенности; интерпретация данных зависит или не зависит от порядка их появления во времени; стоимость приобретения данных; как данные приобретаются (какие вопросы необходимо задать для получения данных); как необходимые характеристики объектов могут быть извлечены из входного сообщения (сигнал, изображение, текст на естественном языке, устная речь и т. п.); данные надежны (ненадежны), точны (неточны); данные согласованы (не согласованы), полные (неполные).

Часто на этапе формализации выясняется, что для различных частей общей задачи нужны различные языки или ИС.

Совет инженеру.

На начальном этапе целесообразнее использовать одно ИС, а не несколько. Это снижает трудоемкость разработки системы.

13.5.4. Этап выполнения. Цель этого этапа — создание одного или нескольких прототипов ЭС, решающих требуемые задачи. Затем на данном этапе по результатам этапов тестирования и опытной эксплуатации создается конечный продукт, пригодный для промышленного использования. Разработка прототипа состоит в программировании его компонентов или выборе их из имеющихся ИС и наполнении БЗ.

Совет инженеру.

Разработку первой версии прототипа ЭС следует начинать, как только будет хорошо понят первый тестовый пример.

Обычная ошибка разработчиков при создании прототипа состоит в том, что процесс приобретения знаний откладывают до полного завершения программирования. Тем самым эта наиболее трудоемкая часть работы отодвигается на поздние этапы, и в процессе накопления знаний приходится вносить изменения в уже готовые программы. Поэтому необходимо начинать приобретение знаний, как только составлены или выбраны программы, позволяющие работать с простейшим представлением знаний и простейшими управляющими структурами, а следовательно, максимально рано начать выполнение отдельных подзадач и обнаружить необходимость дополнительных знаний для их решения. Иными словами, первый прототип экспертной системы — демонстрационный (ЭС-1) должен появиться через несколько месяцев (обычно два-три), а не через годы после начала работы.

Разработка прототипа — чрезвычайно важный шаг в создании ЭС. Некоторые программы прототипа могут войти в окончательную версию ЭС. Однако не это наиболее важная цель создания прототипа. Главное, чтобы прототип обеспечил проверку адекватности идей, методов и способов представления, выбранных при создании данной ЭС, решаемым задачам. Создание первого прототипа должно подтвердить, что выбранные методы решений и способы представления пригодны для успешного решения по крайней мере ряда задач из области экспертизы, а также показать, что с увеличением объема знаний и улучшением стратегий поиска ЭС сможет дать высококачественные и эффективные решения всех задач данной проблемной области. При разработке первого прототипа обычно оставляют в стороне вопросы, требующие значительных трудозатрат: понимание и синтез фраз ограниченного естественного языка; построение сложных моделей; учет сложных временных, причинных и модальных отношений; понимание намерений пользователей (экспертов); моделирование рассуждений, содержащих неточные понятия.

Итак, можно сделать вывод, что в первом прототипе реализуется (используется) простейшая процедура вывода. При его разработке основная цель — получить решение задачи, не заботясь пока об эффективности.

Советы инженеру.

1. Работайте интенсивно с некоторыми представительными подзадачами. Выберите пять-шесть подзадач, охватывающих процесс от входа до выхода и отражающих различные аспекты общей задачи (что позволит ЭС-1 не быть слишком ограниченной). «Извлекайте» из эксперта в первую очередь правила, относящиеся к этим подзадачам.
2. Отделите от ЭС-1 те проблемы, которые трудны для реализации методами инженерии знаний. Примерами таких проблем являются следующие: понимание и генерация ЕЯ-фраз; создание пространственных, временных и причинных моделей; учет модальности; понимание намерений пользователей; рассуждения, использующие «размытые» понятия. Такие проблемы следует решать в рамках ЭС-2.
3. Повсеместно используйте механизм косвенных ссылок, хотя применение их потребует больших усилий, но эти усилия окупаются (например, предпочтительнее использовать в правиле ссылку на имя класса понятий, а не перечислять явно все члены класса).
4. Отделяйте специфические знания, характеризующие конкретную проблемную область, от общих знаний о решении задач, т. е. не употребляйте внутри одного правила знания разных видов (знания о проблемной области и общие знания о механизмах решения). Разделение указанных знаний обеспечивает «прозрачность» системы (например, при генерации объяснений) и упрощает ее развитие.
5. На начальном этапе отдавайте предпочтение простым механизмам вывода, которые легче реализовать и которые обеспечивают легкий доступ к знаниям (в сложных механизмах доступ к знаниям, как правило, оказывается затруднен).
6. Не заботьтесь в начале разработки об экономии времени и памяти ЭВМ. Основная задача — построение БЗ (правил и данных) и выбор или создание механизма вывода, который может решить задачу, поставленную перед данной ЭС.
7. Используйте (если оно существует) или создайте ИС, упрощающее процесс написания правил. Речь идет не о понимании ЕЯ-фраз, а о простых средствах (типах редактора), управляющих и контролирующих процесс ввода или корректировки правил.
8. Уделяйте внимание документации. Несмотря на то, что ЭС-1 является начальным прототипом, целесообразно иметь его описание, отражающее основные характеристики прототипа (принципы построения, методы и т. п.) и процесса его разработки (сроки разработки, участники и т. п.).

После разработки первого прототипа необходимо расширить круг задач, решаемых системой, для того, чтобы собрать пожелания и замечания, которые

будут учтены в очередной версии системы (ЭС-2). Для этого осуществляется развитие ЭС-1 путем добавления: «дружественного» интерфейса; средств для исследования как БЗ, так и цепочек выводов, генерируемых системой (что обеспечивает прозрачность и понимаемость системы разработчиком); средств для сбора замечаний пользователей; средств хранения библиотеки задач, решенных системой. Библиотека необходима для того, чтобы при каждой модификации системы можно было проверить, решаются ли все старые задачи и в новой версии. Расширенная версия ЭС-1 может рассматриваться как исследовательский прототип ЭС (см. § 13.2).

В ходе приобретения знаний инженер по знаниям должен получить знания от эксперта, структурировать их и представить в виде, понятном ЭС. Процесс извлечения знаний сложен и длителен, так как эксперт часто или не осознает, какими знаниями он пользуется, или не может их вербализовать.

Советы инженеру.

1. Не только беседовать с экспертом, но и наблюдать за его работой над решением конкретных задач, выясняя, почему эксперт выполнил (не выполнил) то или иное действие.

2. Использовать термины и методы эксперта (а не свои).

Для достижения эффективного функционирования ЭС необходимо осуществлять структурирование знаний. Наиболее важным средством для этого являются абстрактные понятия промежуточного уровня. Во многих случаях эти понятия могут явно не упоминаться (а возможно, и не осознаваться) экспертом. Инженер по знаниям должен выделить такие понятия, обнаружив сходные действия эксперта при обработке различных ситуаций.

При представлении правил в виде, понятном ЭС, особое внимание следует уделять двум ситуациям: некоторое правило слишком громоздко; имеется много похожих правил. Громоздкость правила может объясняться тем, что в нем отражено несколько фактов из данной проблемной области. Если это так, то правило надо разбить на несколько более мелких. Вторая ситуация возникает тогда, когда в проблемной области существует понятие, явно не указанное экспертом, а возможно, и не имеющее имени. В этом случае новое понятие необходимо ввести в явном виде, присвоить ему специальное имя и, используя это понятие, сформулировать одно правило взамен группы подобных.

Советы инженеру.

1. Попробуйте избежать соблазна (хотя бы на время развития ЭС-1) использовать традиционные программы там, где выбранный формализм неудобен. Помните: программы представляют знания неявно, а цель ЭС — явно представить знания эксперта (именно явность знаний обеспечивает возможность реализации объяснений, приобретения знаний и автоматической генерации решения).

2. Попробуйте сохранить у эксперта интерес к разработке ЭС и веру в конечный успех:

а) объясняя эксперту, что вы совместно с ним решаете чрезвычайно сложную задачу (ошибочно настраивать эксперта на быстрый успех);

б) обеспечивая при расширении ЭС-1 какие-либо возможности, которые предоставляют эксперту непосредственную выгоду от использования ЭС;

в) освобождая эксперта и пользователя от любых технических проблем (использование или изучение программирования или литературы по ЭС и т. п.).

Выполнение экспериментов с расширенной версией ЭС-1, анализ пожеланий и замечаний служат отправной точкой для создания второго прототипа (ЭС-2). Процесс разработки ЭС-2 итеративный. Анализ результатов прогонов тестовых примеров позволяет выявить недостатки ЭС и разработать пути для их устранения. Этот итеративный процесс может продолжаться от нескольких месяцев до нескольких лет в зависимости от сложности проблемной области, гибкости выбранного представления и степени соответствия управляющего механизма решаемым задачам (возможно, потребуется разработка ЭС-3 и т. д.). При разработке ЭС-2, кроме перечисленных задач, решаются следующие:

анализ функционирования системы при значительном расширении БЗ; исследование возможностей системы в решении более широкого круга задач и принятие мер для обеспечения таких возможностей;

анализ мнения пользователя о недостатках систем, о том, какую дополнительную помощь он хочет получить от ЭС, и т. п.;

разработка системы ввода-вывода, осуществляющей анализ или синтез предложений ограниченного естественного языка, что позволяет пользователю взаимодействовать с ЭС-2 в форме, близкой к форме стандартных учебников для данной области.

Если ЭС-2 успешно прошла этап тестирования, то она может классифицироваться как промышленная система (см. § 13.2).

13.5.5. Этап тестирования. В ходе этого этапа осуществляется оценка выбранного способа представления знаний и ЭС в целом. Как только ЭС оказывается в состоянии обработать от начала до конца два или три примера, необходимо начинать проверку на более широком круге примеров, чтобы определить недостатки БЗ и управляющего механизма (процедур вывода). Инженер по знаниям должен подобрать примеры, обеспечивающие всестороннюю проверку ЭС.

Обычно выделяют следующие источники неудач в работе системы: тестовые примеры; ввод-вывод, правила вывода, управляющие стратегии.

Наиболее очевидной причиной неудачной работы ЭС являются недостаточные показательные тестовые примеры. В худшем случае тестовые примеры могут оказаться вообще вне проблемной области, на которую рассчитана ЭС, однако чаще множество тестовых примеров находится в рассматриваемой проблемной области, но является однородным и не позволяет охватить всю проблемную область. При подготовке тестовых примеров целесообразно классифицировать их по подпроблемам предметной области, выделяя стандартные случаи, определяя границы трудных ситуаций и т. п.

Ввод-вывод можно характеризовать данными, приобретенными в ходе диалога с экспертом, и заключениями, предъявленными ЭС в ходе объяснений. Методы приобретения данных могут не давать нужных результатов, так как задавались неправильные вопросы или собрана не вся необходимая информация. Кроме того, вопросы системы могут быть трудными для понимания, многозначными и не соответствующими знаниям пользователя. Ошибки при вводе могут возникнуть из-за неудобного для пользователя входного языка. В ряде приложений для пользователей удобен ввод не только в печатной, но и в звуковой форме, может понадобиться ввод и в графическом виде.

Выходные сообщения (заключения) системы могут оказаться непонятны пользователю (эксперту) либо потому, что их слишком много, либо потому, что их слишком мало. Они могут быть организованы, упорядочены способом, непонятным (неудобным) пользователю, кроме того, выражены на не подходящем для пользователя уровне абстракции с использованием непонятной ему лексики.

Типичным источником ошибок в рассуждениях являются правила вывода. Важная причина здесь — неучет зависимостей между правилами (точнее, между их означиваниями). Дело в том, что правила редко независимы друг от друга, хотя на первом этапе отладки удобно считать их таковыми. Среди других причин ошибок можно отметить ошибочность, противоречивость и неполноту правил. Если неверна посылка правила, то это может привести к использованию правила в неподходящем контексте. В данном случае система не замечает частных подслучаев. Если ошибочно действие правила, то конечный результат трудно предсказать. Правило может быть ошибочно, если при корректности его условия и действия нарушено соответствие между ними. Так как правила зависят друг от друга, важно учесть возможность влияния одних правил на другие. Контекстные зависимости следует учитывать, в частности, при расстановке различных коэффициентов, характеризующих правила. Необходимо принять меры против порождения гипотез (промежуточных заключений), которые правдоподобны каждая в отдельности, но комбинируются в бессмысленные

последовательности. Требуется разработать специальные правила, препятствующие образованию ошибочных комбинаций.

Весьма часто к ошибкам в работе ЭС приводят управляющие стратегии. Возможно, изменение стратегии необходимо, если ЭС рассматривает сущности в порядке, отличном от «естественного» для эксперта. Последовательность, в которой данные рассматриваются ЭС, не только влияет на эффективность работы системы, но и может приводить к изменению конечного результата. Например, рассмотрение правила X до правила Y может иногда привести к тому, что правило Y всегда будет игнорироваться системой. Изменение стратегии необходимо и в случае неэффективной работы ЭС. Кроме того, недостатки в управляющих стратегиях могут привести к чрезмерно сложным заключениям и объяснениям ЭС.

Критерии, с помощью которых оценивается ЭС, зависят от того, с чьей точки зрения дается оценка. Например, при тестировании ЭС-1 оценка осуществляется с точки зрения эксперта, для которого важна в первую очередь полнота и безошибочность правил вывода. При тестировании промышленной системы оценка производится в основном с точки зрения инженера по знаниям, которого интересует эффективность работы ЭС. При тестировании ЭС после опытной эксплуатации оценка осуществляется с точки зрения пользователя, заинтересованного в первую очередь в удобстве работы и получении практической пользы.

Советы инженеру.

1. Как можно раньше определите, как вы будете оценивать успех (неудачу) ваших усилий, т. е. что является вашей целью: «тиражирование» знаний эксперта, успешное решение новых, ранее не решавшихся экспертом задач и т. п.

2. Как можно раньше определите, как эксперт будет оценивать качество системы: достаточно ли просто получение правильного ответа или требуется также «человеческий» способ его получения.

3. Как можно раньше определите, как эксперт (пользователь) будет оценивать удобство системы и ее полезность.

13.5.6. Этап опытной эксплуатации. На данном этапе проверяется пригодность ЭС для конечного пользователя. Здесь ЭС занимается решением всех возможных задач при работе с различными пользователями. Целесообразно организовать работу ЭС не на стенде разработчика, а на месте работы пользователей. К этому этапу следует переходить лишь после того, как система, по мнению эксперта, будет успешно решать практически все требуемые задачи, чтобы ошибки в решениях не создавали у пользователя отрицательного представления о ЭС. Пригодность ЭС для пользователя определяется в основном удобством работы с ней и ее полезностью. Под *полезностью* ЭС понимается способность ее в ходе диалога определять потребности пользователя, выявлять и устранять причины неудач в работе, а также удовлетворять потребности пользователя (т. е. решать поставленные задачи). Другими словами, пользователю важно довести «до сознания» ЭС свою информационную потребность несмотря на возможные ошибки, допускаемые им из-за недостаточного знания ЭС. Конечно, для пользователя важны также полнота и правильность решений, но эти характеристики должны быть проверены экспертом на предыдущем этапе.

Под *удобством работы* ЭС понимается естественность взаимодействия с ней (общение в привычном, не утомляющем пользователя виде), гибкость ее (способность системы настраиваться на различных пользователей, а также учитывать изменения в квалификации одного и того же пользователя) и устойчивость системы к ошибкам (способность не выходить из строя при ошибочных действиях неопытного пользователя).

По результатам эксплуатации может потребоваться не только модификация программ и данных (совершенствование или изменение языка общения, диалоговых средств, средств обнаружения и исправления ошибок, настройка на пользователя и т. д.), но и изменение устройств ввода-вывода из-за их не-

приемлемости для пользователя. По результатам этого же этапа принимается решение о переносе системы на другие ЭВМ (например, для расширения сферы использования ЭС и (или) снижения ее стоимости).

После успешного завершения этапа опытной эксплуатации и использования различными пользователями ЭС может классифицироваться как коммерческая (см. § 13.2).

13.5.7. Модификация системы. В ходе создания ЭС почти постоянно осуществляется ее модификация. Можно выделить следующие виды модификации системы: переформулирование понятий и требований, переконструирование представления и усовершенствование прототипа (см. рис. 13.2).

Усовершенствование прототипа осуществляется в процессе циклического прохождения через этапы выполнения и тестирования для отладки правил и процедур вывода. Циклы повторяются до тех пор, пока система не будет вести себя ожидаемым образом. Изменения, осуществляемые при усовершенствовании, зависят от выбранного способа представления и класса задач, решаемых ЭС. Если в процессе усовершенствования желаемое поведение не достигается, то необходимо осуществить более серьезные модификации архитектуры системы и БЗ.

Возврат от этапа тестирования на этап формализации приводит к пересмотру выбранного ранее способа представления знаний. Данный цикл называют *переконструированием*.

Если возникшие проблемы еще более серьезны, то после неудачи на этапе тестирования может потребоваться возврат на этапы концептуализации и идентификации. В этом случае речь будет идти о *переформулировании* понятий, используемых в системе, — о проектировании всей системы практически заново.

13.6. Взаимодействие инженера по знаниям с экспертом

Мощность ЭС как систем, основанных на знаниях, зависит в первую очередь от качества и количества знаний, хранимых в них. Поэтому ясно, что процесс приобретения знаний для ЭС наиболее важный. Так как в настоящее время не существует методов автоматического приобретения знаний, процесс наполнения ЭС знаниями является весьма трудоемким. Знания для ЭС могут быть получены из различных источников (книг, отчетов, баз данных, эмпирических данных, персонального опыта эксперта, инженера и т. п.). Однако наиболее значимые знания в настоящее время приобретаются от людей-экспертов.

Извлечение знаний из эксперта (экспертов) осуществляется в процессе интенсивного систематического (в течение нескольких месяцев) взаимодействия инженера по знаниям с экспертом. Сложность извлечения знаний состоит в том, что, как правило, эксперт либо не осознает, какими знаниями он пользуется для получения решения, либо не может сразу эти знания вербализовать. Поэтому инженер по знаниям должен работать с экспертом в контексте решения конкретных задач (подзадач). Обычно оказывается неэффективным непосредственно спрашивать эксперта о том, с помощью каких методов он решает ту или иную задачу. В этом случае проявляется парадокс экспертизы (инженерия знаний): чем выше компетентность эксперта, тем меньше его способность описать знания, используемые им для решения задач. Более того, анализ попыток экспертов объяснить, как они формируют решение задач, показывает, что они часто описывают правдоподобные линии рассуждений, мало похожие на те, которыми они действительно пользуются. Из этих наблюдений сделаны два следующих важных вывода для построения ЭС:

1. Не будьте сами себе экспертом! (другими словами, если вы эксперт, то не пытайтесь описать свои знания без инженера по знаниям, который должен убедиться в достоверности выделенных знаний).

2. Инженер по знаниям должен верить не тем знаниям, за которые эксперт ручается, а тем знаниям, которые он действительно использует для решения задач.

Исследование работы экспертов базируется на методах психологии: наблюдения и интуитивных. В первом случае исследование проводится в процессе решения задачи инженером по знаниям, а во втором — самим экспертом (интуитивные методы базируются на интроспекции — самонаблюдении).

Метод наблюдения, как правило, реализуется следующим образом. Эксперта при решении задачи просят «думать вслух». Раздумья эксперта оформляются в виде протокола, который затем исследуется инженером по знаниям (с возможным привлечением эксперта). Протокол может дать инженеру некоторую информацию об организации, содержании знаний эксперта и о том, какие механизмы вывода использует эксперт. Метод наблюдения может дополняться фазой «очищения», в ходе которой эксперт (эксперты) комментирует знания, выделенные в ходе наблюдения. Исследователи, применяющие этот метод, обычно не прерывают эксперта (вопросами или комментариями) в ходе решения им задачи.

Интуитивный метод применяется в одной из двух модификаций. В первом случае инженер по знаниям изучает литературу и взаимодействует с экспертом для того, чтобы овладеть данной проблемной областью, т. е. он становится псевдоэкспертом. Действуя подобным образом, инженер формирует знания, описывающие данную область, и затем проверяет выделенные знания, взаимодействуя с другими экспертами. Во втором случае сам эксперт действует как «строитель» теории своего собственного поведения, используя интроспекцию.

При использовании обоих методов (наблюдения и интуитивного) исследователь сталкивается с серьезными трудностями. В методе наблюдения эксперт решает задачу и одновременно описывает то, что он делает. «Думание вслух» может изменить технику эксперта, а кроме того, при описании процесса решения эксперт может допускать (и допускает) существенные пропуски. Инженер по знаниям может сталкиваться с серьезными затруднениями при попытке заполнить эти пропуски (даже если он привлекает эксперта). Эксперт (при анализе протокола) может восполнить пропуск в процессе рассуждения, однако нет гарантии, что предложенный им ход рассуждений отражает его рассуждения при решении задачи. При использовании интуитивных методов исследователь также сталкивается с определенными трудностями. Эксперт часто получает решение, не осознавая, какие знания и механизмы он использует. В этом контексте ясно, что интроспекция может не отражать реальных действий эксперта.

Инженер по знаниям может использовать тот и другой метод, он должен играть активную роль в процессе извлечения знаний, задавать вопросы, предлагать понятия, правила и гипотезы. Некоторые приемы по извлечению знаний из эксперта приведены в табл. 13.1.

Первый прием (наблюдение) позволяет инженеру составить некоторое представление о сложности задач и видах взаимодействия, используемых экспертом при решении задачи. Для ряда проблемных областей наблюдение оказывается малополезным приемом, так как многие аспекты решения остаются непонятными.

Цель второго приема — определить способ организации знания в каждой задаче, способ представления понятий и гипотез, а также несоответствий, неточностей данных (правил). При этом уместно ответить на следующие вопросы: чем рассматриваемая задача отличается от других типичных задач; какие данные необходимы для рассматриваемой задачи; какого вида решения соответствуют рассматриваемой задаче; можно ли задачу свести к подзадачам, не требующим взаимодействия с пользователем; какие правила необходимы для решения данной задачи; какие объяснения (оправдания) уместны для данной задачи?

Третий прием помогает инженеру определить типичную задачу для каждого типа ответа и может помочь выбрать стратегию или основной подход для решения данной задачи. Этот прием успешно работает при рассмотрении диагностических задач.

Приемы	Списание
1. Наблюдение	Инженер наблюдает (не вмешиваясь) за тем, как эксперт решает реальную задачу
2. Обсуждение задачи	Инженер на представительном множестве задач неформально обсуждает с экспертом данные, знания и процедуры решения
3. Описание задачи	Эксперт описывает типичные задачи для основных типов ответов
4. Анализ задачи	Эксперт решает «вслух» реальные задачи, детализируя ход рассуждений
5. Проверка системы (прототипа)	Эксперт предлагает инженеру перечень задач для решения (от простых до сложных), которые (с использованием приобретенных знаний) перед решением системы инженер решает вручную
6. Исследование системы	Эксперт исследует и критикует правила и механизм вывода системы
7. Оценка системы	Инженер предлагает другим экспертам оценить решения разработанной системы (прототипа) и решения эксперта, наполнявшего систему

В ходе четвертого приема эксперт, решая «вслух» задачи, должен стремиться предельно детализировать ход своих рассуждений. Инженер затем, задавая вопросы к каждому шагу рассуждений, стремится выявить гипотезы, стратегии и цели, преследуемые экспертом. Обычно при анализе рассуждений эксперта инженер использует стратегию «поиск в глубину», т. е. он спрашивает эксперта, как достигнуто заключение X, как получено заключение, на котором базируется X. При выявлении правила инженер стремится обобщить его, т. е. сформулировать в предельно общей форме, не нарушающей, однако, его способность приводить к решению. Для определения корректности операций обобщения может потребоваться помощь эксперта.

Решение инженером тестовых задач, предложенных экспертом, до появления прототипа ЭС способствует проверке полноты и согласованности извлеченных знаний. Как только прототип ЭС сможет работать (хотя бы с ограниченными возможностями), его надо использовать для решения тестовых задач.

В ходе исследования системы эксперт проверяет точность каждого правила и его оправдания, используемые при формировании объяснений. Эксперт сравнивает стратегии прототипа со своими, применяемыми при решении задачи.

В процессе оценки системы (прототипа) инженер сравнивает стратегии различных экспертов и пытается выделить существенные различия. Чем выше искусство инженера по знаниям, тем больше окончательный вариант ЭС будет отражать эвристики и процедуры, используемые экспертом при решении задачи.

13.7. Трудности разработки экспертных систем

При разработке ЭС разработчиков поджидают различные трудности: глобальные, имеющие отношение ко всему процессу разработки или к нескольким этапам разработки, и локальные, проявляющиеся в основном на некотором этапе разработки.

13.7.1. Глобальные трудности. Это общие проблемы, с которыми сталкивается коллектив, пытающийся применить технологию ЭС для решения своих задач.

Источником глобальных трудностей являются по крайней мере следующие факторы: недостаток ресурсов; ограничения существующих ЭС; длительность разработки ЭС. Рассмотрим эти факторы подробнее.

Недостаток ресурсов для создания ЭС выражается в нехватке человеческих, программных и аппаратных ресурсов. В связи с относительной новизной направления практически отсутствуют инженеры по знаниям, очень мало специалистов по разработке ИС для систем ИИ и ЭС.

Ограничения программных ресурсов выражаются в слабом развитии ИС, что приводит к чрезмерной длительности и трудоемкости разработки ЭС.

Ограничения аппаратных ресурсов выражаются в отсутствии в СССР символьных ЭВМ, ориентированных на разработку ЭС и систем ИИ. На Западе последовательные символьные ЭВМ (типа ЛИСП-машин) серийно выпускаются с 1981 г. С 1985 г. в США начат серийный выпуск параллельных символьных ЭВМ.

Ограничения существующих ЭС можно разделить на ограничения, присущие существующим ИС, и на ограничения, присущие собственно ЭС. К ограничениям собственно ЭС следует отнести следующие: крайне слабые возможности по работе с динамическими мирами, требующими представления времени и (или) пространства; недостаточные возможности ЭС по генерации умозаключений, основанных на «здоровом смысле»; ЭС плохо распознают ситуации, когда пользователь выходит за рамки их возможностей (что приводит к непредвиденным неудачам); как правило, ЭС не способны определить несогласованность (противоречивость) знаний, объяснить причину несогласованности и устранить ее. К ограничениям ИС можно отнести следующее: они могут оказать очень малую помощь в процессе приобретения и корректировки знаний в наиболее трудоемком аспекте разработки ЭС; существующие отечественные ИС, как правило, не позволяют использовать смешанное представление (например, правила и фреймы).

Процесс разработки ЭС довольно длительный, и это является существенным ограничением при выборе данного способа реализации задачи.

13.7.2. Локальные трудности. Инженер по знаниям на разных этапах создания ЭС может столкнуться с локальными трудностями.

На этапе идентификации основные трудности возникают при решении следующих проблем.

1. Определение пригодности методов инженерии знаний для решения предлагаемой задачи. При решении данной проблемы могут возникнуть следующие препятствия:

задача слишком сложна и не может быть решена доступными ресурсами в пределах заданных ограничений (время, тип ЭВМ и т. п.);

задача может быть решена ЭС, но пользователь не получит от ее решения существенной пользы;

задача может быть решена, но либо требуется ввести слишком много правил и объектов (что чрезмерно затянет процесс разработки), либо результирующая система работает слишком медленно.

Чтобы выявить сложность задачи, инженер должен разработать небольшой прототип ЭС, работа которого позволит надежно ответить на вопрос о приемлемости задачи. В простых случаях ответ на этот вопрос можно получить, имея опыт разработки ЭС, при анализе описания задачи. Для выявления полезности ЭС следует до ее разработки ответить на следующий вопрос: если ЭС будет работать хорошо, то будет ли это давать тот эффект, которого хочет достичь пользователь? Другими словами, надо понять, не окажется ли, что для удовлетворения потребностей пользователя надо решать еще какие-то задачи, не предусмотренные в данной ЭС. Для преодоления последнего из перечисленных препятствий необходимо ограничить задачу или область ее приложения. Надежным средством выявления наличия или отсутствия данной трудности является разработка прототипа.

2. Определение необходимых ресурсов. При решении этой проблемы возникают следующие препятствия: срок разработки ЭС назначается на основании

не анализа, выполненного инженером по знаниям, а потребностей пользователей (интерактивная природа процесса приобретения знаний существенно ограничивает эффективность ускорения разработки ЭС за счет увеличения числа разработчиков); попытка разработать ЭС без инженеров по знаниям, а только с помощью программистов приводит либо к чрезмерному увеличению сроков разработки, либо к ее полной неудаче.

3. Выбор эксперта. При работе с экспертом может оказаться, что процесс извлечения знаний протекает чрезвычайно неэффективно. Наиболее вероятной причиной этого обычно является неправильный выбор эксперта. При выборе эксперта к нему следует предъявлять следующие требования: высокая компетентность в данной проблемной области; способность ясно, просто и доходчиво выражать свои идеи и методы; наличие у эксперта заинтересованности в разработке ЭС; эксперт должен верить в полезность использования вычислительной техники при решении данной задачи (чем больше эксперт знает об ЭВМ и программировании, тем лучше); эксперт должен хотеть и быть способен уделять разработке ЭС примерно половину своего рабочего времени в первые полгода разработки и примерно одну треть времени в последующее время.

На этапе концептуализации могут встретиться следующие трудности:

1. К работе подключено излишне много экспертов. Инженер по знаниям, общаясь со всеми, не имеет возможности глубоко исследовать данную область. Инженеру рекомендуется работать только с одним или двумя экспертами. После построения прототипа для его тестирования и модификации целесообразно подключать дополнительных экспертов.

2. Инженер по знаниям выявляет правила, наблюдая за тем, как эксперт решает конкретные задачи, что может привести к образованию специфических правил. Для преодоления этой трудности инженер должен пытаться выявить общие принципы, используемые экспертом. Часто оказывается полезным введение понятий высокого уровня, которые эксперт фактически использует, но не называет, так как они им не вербализованы (в научной литературе такие понятия не введены). Введение этих понятий может позволить заменить несколько специфических правил одним общим.

3. После нескольких месяцев взаимодействия с экспертом инженер по знаниям выделил несколько сотен правил и представил их в выбранном языке. Однако начало проверки прототипа показывает, что многие фундаментальные понятия в приобретенных правилах отсутствуют (т. е. выявлены ошибки, допущенные на этапе концептуализации). Чтобы описанная выше ситуация не возникла, опытный инженер по знаниям осуществляет тестирование принятых решений в ходе всех этапов разработки, а не только по окончании этапа выполнения.

На этапе формализации основные проблемы относятся к выбору и использованию ИС. Можно выделить следующие трудности.

1. При использовании выбранного ИС инженеры по знаниям (разработчики) приходят к выводу, что в нем трудно представлять как понятия области экспертизы, так и управляющие структуры, необходимые для решения задачи. Выбор ИС и определение его пригодности для данного приложения весьма сложны и неформальны. Поэтому разработчики, как правило, могут оценить сделанный выбор только после разработки в нем небольшого прототипа. Если разработчики приходят к выводу, что выбранное ИС не подходит для данной проблемы, то необходимо либо выбрать другое ИС, либо разработать новое (в этом случае практически дело сводится к откладыванию разработки ЭС на неопределенный срок).

2. Инженер по знаниям выбирает то ИС, которым он хорошо владеет, а не то, которое наиболее соответствует данной проблемной области. При этом, если ИС явно плохо подходит, инженер прилагает усилия к изменению исходной задачи (что может снизить или устранить практическую значимость создаваемой ЭС), чтобы она подходила под выбранное им ИС. Для устранения подобных затруднений руководитель проекта по разработке ЭС должен при

выборе ИС проконсультироваться с несколькими инженерами по знаниям, чтобы убедиться в обоснованности сделанного выбора. Кроме того, сам инженер по знаниям должен осознавать наличие подобной ловушки и тщательно подходить к проблеме выбора ИС.

3. В качестве ИС выбирается (на начальных этапах разработки ЭС) один из общецелевых языков программирования (например, Паскаль, Си, Фортран), что приводит к чрезмерному увеличению времени разработки ЭС или к полной неудаче всего проекта. Ошибочность подобного выбора состоит в том, что для выполнения таких наиболее трудоемких работ, как наполнение ЭС знаниями, тестирование и корректировка БЗ, общецелевые языки плохо приспособлены. Более правильно разработать ЭС, используя ИС, ориентированные на разработку ЭС, и затем, если ЭС будет работать медленно, переписать ее, используя общецелевые языки программирования.

4. Выбранное ИС содержит ошибки, препятствующие использованию многих из его свойств. Для предотвращения указанной трудности рекомендуется: избегать ИС, не находившихся в употреблении; не использовать ИС, которые более не сопровождаются разработчиком; выбирать то ИС, которое сопровождается разработчиком и было успешно использовано в других применениях.

В ходе этапа выполнения могут встретиться следующие основные трудности:

1. Эксперт теряет интерес к разработке ЭС. Время, которое он выделяет на работу с инженером по знаниям, постоянно сокращается. В подобных ситуациях рекомендуется: поддерживать интерес эксперта к работе, обеспечивая дружественную обстановку; объяснять, что для успеха необходимы регулярные контакты (не реже 2—3 раз в неделю); помнить, что эксперт не специалист в программировании и ЭС, и не требовать от эксперта того, что он не умеет делать; предоставить эксперту возможность непосредственного наблюдения за результатами предлагаемых им изменений БЗ.

2. Эксперт при корректировке БЗ прототипа не совсем понимает, как система интерпретирует правила. Чтобы избежать указанного затруднения, рекомендуется: использовать в правилах ту же терминологию, которую использует эксперт; хранить в системе определения всех используемых понятий и иметь возможность выдавать их по требованию пользователя; если правила во внутреннем представлении «нечитаемы», то иметь средства для преобразования правил в вид, понятный пользователю.

3. Правила, введенные экспертом, в сложных ситуациях не позволяют прототипу ЭС получить качественные решения. Для преодоления данного затруднения рекомендуется: ориентировать эксперта на решение реальных, а не «игрушечных» задач; наблюдать за работой эксперта, стремясь выделить все типы возможных задач (подзадач); привлекать для оценки работы прототипа других экспертов.

4. Для построения ЭС было выбрано ИС без объяснительных способностей. После отработки ЭС разработчики пытаются дополнить ЭС этим механизмом, однако обычно сделать это чрезвычайно трудно. Чтобы не сталкиваться с подобными трудностями, разработчики должны выбирать ИС с объяснительными способностями. Объяснения ускоряют отладку, тестирование, модификацию прототипа и обеспечивают удобство для конечного пользователя и эксперта.

5. В процессе разработки ЭС знания об области экспертизы переплетаются с общими знаниями о решении задач (проблемно-независимыми знаниями), что затрудняет (делает невозможным) проведение модификации и развития ЭС. По сути дела, нарушается основной принцип разработки ЭС — отделение знаний эксперта от остальных знаний и представление знаний эксперта в явном виде. Соблюдение этого принципа упрощает развитие системы и выполнение объяснений, а в ряде случаев позволяет обеспечить проверку непротиворечивости правил и данных.

На этапе тестирования и опытной эксплуатации могут встретиться следующие трудности.

1. Тестирование не выполняется в ходе разработки, а впервые осуществляется после этапа выполнения. При подобном подходе вероятность отрицательной оценки ЭС очень велика. Кроме того, весьма вероятно, что перепроектирование ЭС придется начать с ранних этапов (идентификации и концептуализации). Общая рекомендация при разработке ЭС состоит в проведении тестирования и оценки как можно раньше. Поэтому на начальных этапах разработки ЭС должны составить методику оценки пригодности и полезности конечного продукта, что позволит по ходу разработки (а не в ее конце) оценивать состояние проекта и прогнозировать срок его успешного завершения.

2. Пользователи (эксперты) находят процесс взаимодействия с ЭС трудным и неудобным: им не понятны сообщения системы, время реакции ее велико, возможности неочевидны и трудноиспользуемы и т. п. Для устранения подобных ситуаций целесообразно освободить пользователей от любой технической работы, непосредственно не связанной с областью экспертизы (работа с операционной системой, сопровождение словарей, взаимодействие с «недружественным» редактором). Недопустимо использовать аббревиатуры. При проектировании необходимо предусмотреть то, что с ЭС работают различные пользователи (конечные пользователи, эксперты, инженеры и т. п.).

3. Если правил больше нескольких сотен, возникает затруднение с дальнейшей корректировкой и добавлением их, так как внесение одних изменений порождает цепь других, процесс перестает быть управляемым и кажется бесконечным. Выход из подобных ситуаций — разработка системы сопровождения, запинающей, какие задачи решены, какие получены результаты, какие правила использованы, когда и какие изменения внесены и т. п. Целесообразно создать множество тестовых задач, проверяющих работоспособность системы после внесения изменений.

13.8. Проблемы и перспективы

Несмотря на то, что ЭС широко используются в США, Японии и Европе, существует ряд нерешенных проблем, препятствующих их массовому производству и распространению.

1. Разработка ЭС до сих пор весьма длительный и трудоемкий процесс. В этом процессе наиболее узким местом является приобретение знаний, включающее извлечение, структурирование, представление, отладку (обеспечение полноты, непротиворечивости знаний, гарантия качества решений и т. п.) и сопровождение знаний. Эта проблема усугубляется тем, что существующие ИС и оболочки поддерживают не все этапы разработки ЭС, а только этапы формализации и отчасти выполнения и тестирования. При этом ранние наиболее неформальные этапы (идентификации и концептуализации) оказываются практически не поддержаны существующими ИС. Кроме того, даже лучшие ИС типа KEE и ART предлагают (на этапе формализации) инженеру по знаниям множество возможностей, ничего не говоря о том, какую из них выбрать.

Проблема приобретения знаний усугубляется тем, что без постоянного обслуживания и совершенствования экспертами сложные ЭС теряют (в связи с изменением окружения) эффективность и точность предлагаемых решений.

2. Большинство существующих ЭС и ИС, с использованием которых эти ЭС созданы, базируются на программных средствах типа Лисп (иногда Пролог), трудно согласуемых с программами, составленными на традиционных языках программирования. Указанное обстоятельство затрудняет создание гибридных систем на традиционных ЭВМ. Подобные затруднения отсутствуют при использовании символьных ЭВМ. Однако их высокая стоимость препятствует массовому распространению ЭС.

3. Подавляющее большинство ЭС решает статические задачи в статических предметных областях. Однако многие (если не большинство) важные практические приложения являются динамическими. В настоящее время мало ИС, ориентированных на создание динамических приложений. Более того, слабо

развита теория представления и решения динамических приложений. Мало параллельных символьных ЭВМ, предназначенных для решения динамических задач в изменяющемся окружении.

Массовому распространению ЭС в СССР кроме перечисленных выше проблем препятствуют следующие обстоятельства:

1. Мало разнообразных ИС. Потенциальные заказчики (а иногда и разработчики) не понимают, что ИС не являются универсальными. Даже оболочки ЭС целесообразно ориентировать на решение только конкретных типов задач (например, оболочки для задач диагностики, распознавания и т. п.). Кроме того, необходимы ИС, использующие как разнообразные способы представления знаний, вывода, приобретения знаний, объяснения, так и ИС, предпочтительно ориентированные на различные стадии существования ЭС.

2. Недостаточно инженеров по знаниям и практически отсутствует как подготовка студентов, так и переподготовка инженеров.

3. Создание простых ЭС сдерживается малым количеством ПЭВМ, а создание сложных ЭС — отсутствием символьных ЭВМ.

4. Отсутствуют надежные механизмы, позволяющие заинтересовать эксперта работой по созданию ЭС.

5. В большинстве случаев отечественные эксперты (когда они найдены) предлагают создавать ЭС для решения не тех неформализованных задач, в которых они являются специалистами (так как решение этих задач лично им обычно мало что дает и, более того, при успехе создает в лице ЭС потенциального конкурента), а тех, которые им надо решать, но они не умеют этого делать. Таким образом, эксперты, чрезмерно уверовав в возможности инженерии знаний, предлагают создавать ЭС для тех приложений, где нет экспертов, обречая тем самым работу на неудачу.

6. Большинство ИС и ЭС в СССР до сих пор разрабатывают академические и учебные учреждения, которые, как правило, не заинтересованы в создании промышленных и коммерческих ЭС, да и вряд ли способны их создавать (в связи со значительной трудоемкостью). Таким образом, публикаций об ЭС и ИС в СССР много, а практически использовать оказывается иногда и нечего.

7. Отсутствуют предприятия, осуществляющие консультации по созданию ЭС и по обучению работе с конкретными ИС.

Перспективы развития ЭС в основном связаны с решением перечисленных выше проблем.

С целью резкого сокращения сроков и снижения стоимости создания ЭС разрабатываются различные ИС, что позволит, по мнению зарубежных специалистов, уже к 1990 г. сократить затраты на разработку ЭС примерно в 10 раз. Основу таких средств составят специализированные оболочки (и ИС) и системы создания и поддержания БЗ.

Специализированные оболочки и ИС (в отличие от универсальных оболочек в ИС) ориентируются на определенный тип приложений (например, оболочка для диагностики технических средств Micro In Ate, ИС для моделирования боя Symtactics, оболочка для финансовых приложений FFAST). Специализированные оболочки и ИС (по сравнению с универсальными) являются более эффективными и, кроме того, благодаря специализации могут содержать общие сведения о данном классе приложений, что позволяет существенно упростить, ускорить и удешевить процесс приобретения знаний.

Системы создания и поддержания БЗ предназначены для автоматизации процесса приобретения знаний на всех этапах разработки ЭС. Подобная система обычно ориентируется на класс ИС (в первую очередь, на класс оболочек ЭС). По этой причине, в отличие от оболочек ЭС или, как иногда говорят, *оболочек применения*, эти системы называют *оболочками приобретения знаний*. Примерами таких систем являются NeoETS (ориентированная на создание БЗ для KEE, S1, M.1, OPS-5 и др.), KEATS, TDE.

Устранение трудностей, возникающих при создании гибридных систем, объединяющих ЭС с традиционными программами, осуществляется следующими путями: создание интегральных ИС, органично сочетающих методы инжене-

рии знаний с методами традиционного программирования; создание ИС для ЭС не на базе символьных языков, а на базе языков традиционного программирования (например, Си, Ассемблер); повышение конкурентоспособности символьных ЭВМ (по сравнению с ПЭВМ) за счет существенного снижения их стоимости.

С целью создания динамических ЭС осуществляются следующие мероприятия. Во-первых, начинают появляться первые оболочки и ИС, предназначенные для работы в системах реального времени: Picon (Lisp Machine Jns., США), G2 (Gensym Corp., США). Появление таких средств позволяет резко расширить сферу применения ЭС и начать их использование в ранее недоступных областях: комплексная автоматизация производства, автоматизированные испытания, управление промышленным производством, операции на рынке ценных бумаг, робототехника и т. п. Во-вторых, в рамках ЭС происходит объединение программ, знаний, ЭВМ и специального оборудования, обеспечивающего как восприятие и анализ изменяемого окружения (например, сигнальные ЭВМ и нейронные сети), так и воздействие на него. В-третьих, расширяются исследования и производство параллельных символьных и сигнальных ЭВМ.

Глава 14.

Состояние разработки инструментальных средств и экспертных систем

Б. С. Кирсанов, Э. В. Попов

14.1. Характеристики инструментальных средств и экспертных систем

Данная глава посвящена обзору инструментальных средств (ИС) для создания ЭС. Рассмотрение проводится в соответствии с характеристиками, введенными в § 13.2, 13.3.

14.1.1. Характеристики ИС. Перед тем как перейти к анализу конкретных ИС, приведем обобщенную картину состояния ИС во введенных ранее терминах и там, где это потребуется (для различения ИС), детализируем введенные характеристики.

Большинство ИС предназначено для создания прототипов ЭС, решающих статические задачи (обычно задачи расширения) в статических проблемных областях.

По степени отработанности ИС выделяют следующие стадии существования: исследовательская, промышленная, коммерческая. Так как для пользователя наибольший интерес представляют ИС, достигшие коммерческой стадии (в связи с наибольшей их отработанностью), здесь будут в основном рассмотрены эти ИС.

Различают следующие типы ИС: 1) языки программирования; 2) языки инженерии знаний; 3) средства автоматизации разработки (проектирования) ЭС; 4) оболочки ЭС. Тип является интегральной характеристикой, идентифицирующей большую часть свойств ИС. С точки зрения потребителя, на выбор ИС влияют аспекты: затраты труда на построение ЭС или ее прототипа с помощью ИС; эффективность функционирования ЭС, построенной на основе выбранного ИС; квалификация разработчика, необходимая для применения ИС. Указанные выше типы ИС перечислены в порядке убывания эффективности и требуемой квалификации пользователя и в порядке возрастания трудозатрат, необходимых для создания программного обеспечения ЭС. Поэтому прототипы ЭС создаются с помощью оболочек или средств автоматизации разработки, так как основная цель на этой фазе — минимизация трудозатрат, а промышленные

или коммерческие ЭС — на языках инженерии знаний или языках программирования, обеспечивающих более высокую эффективность.

Оболочки ЭС ориентированы на работу с пользователем — непрофессионалом в области программирования. Основным свойством оболочек является то, что они содержат все компоненты ЭС в готовом виде и их использование не предполагает программирования, а сводится лишь к вводу в оболочку знаний о проблемной области. Каждая оболочка характеризуется фиксированным способом представления знаний и организации вывода и функционирования компонентов, которые будут использоваться во всех приложениях, где будет применяться оболочка. Наиболее популярные оболочки обладают следующими свойствами: 1) решают задачи класса расширения в статических предметных областях в условиях ненадежности знаний; 2) представляют процедурные знания в виде правил; 3) описывают предметную область в виде значений неструктурированных переменных и утверждений, снабженных мерой их истинности (определенности).

Желание предоставить разработчику ЭС разнообразные средства для учета особенностей приложения привело к объединению в рамках одной системы различных методов решения задач, представления и интерпретации знаний. В их состав могут входить средства модификации функционирования оболочки, набор компонентов, позволяющих конструировать собственные оболочки, средства комплексирования компонентов в виде языка высокого уровня, развитые графические интерактивные средства общения с пользователем. Подобные средства называют средствами автоматизации проектирования (разработки) ЭС. Они обычно позволяют представлять предметную область в виде сложной структуры, состоящей из произвольного числа понятий. Необходимость выбора средств, анализа их возможностей, модификации их функционирования и т. д. требует от пользователя специальных знаний в области искусственного интеллекта. Поэтому продажа подобных средств автоматизации проектирования ЭС, как правило, сопровождается обучением потребителей на основе специально разработанных курсов, включающих основы инженерии знаний и особенности использования конкретных средств. Стоимость обучения может соответствовать или превышать иногда значительно (несколько десятков тысяч долларов) стоимость самих средств проектирования ЭС.

По типу используемых методов и знаний большинство ИС, разработанных на ПЭВМ, являются традиционными, т. е. используют только методы и способы инженерии знаний. Однако в последнее время начинают появляться гибридные ИС для ПЭВМ (например, GURU, ЭКСПЕРТ). ИС, разработанные для символьных ЭВМ, интеллектуальных рабочих станций и ЭВМ общего назначения, как правило, являются гибридными (см., например, КЕЕ, ЭКСПЕРТИЗА).

Характеристика «Универсальность» определяет возможность ИС в использовании различных способов представления знаний в рабочей памяти и базе знаний и различных парадигм функционирования системы. Наличие универсальности позволяет адекватно отображать в системы различные типы знаний о проблемной области, добываясь их «объясняемости», прозрачности, легкости приобретения и модификации. Универсальностью могут в той или иной мере характеризоваться как оболочки, так и средства автоматизации проектирования ЭС. Однако в оболочках способ использования различных методов инженерии знаний фиксирован, а в средствах проектирования их использованием управляет инженер по знаниям. К настоящему времени в большинстве ИС при представлении знаний используют фреймы и сети, а в качестве механизма функционирования, как правило, программирование, ориентированное на правила.

Характеристика «Основные свойства» определяет особенности, которые присутствуют в реализации основных программных компонентов системы. Для решателя наиболее важен способ сопоставления и основной способ планирования вычисления (построение цепочек вывода от данных или от целей).

Задача сопоставления состоит в том, чтобы определить, какое из правил, хранящихся в БЗ, может быть применено к текущему состоянию предметной области, хранимой в РП. Сложность сопоставления состоит в том, что в об-

шем случае в каждый момент времени одновременно может быть применено несколько правил и каждое правило к нескольким различным данным, описывающим текущее состояние предметной области. Способы сопоставления, допускаемые ИС, в значительной мере определяют общность решателя и выразительность используемого языка представления знаний. Способы сопоставления в значительной мере зависят: от типа ссылки на объекты РП, используемого в правиле; вида данных РП, сопоставляемых со ссылками; вида проверок, выполняемых в ходе сопоставления.

Выделяют следующие типы ссылок:

конкретная, когда ссылка (идентификатор) в условии правила является адресом конкретного элемента РП;

абстрактная, когда ссылка (идентификатор и его описание) в условии правила именует не конкретный, а любой элемент в РП, свойства которого сопоставляются с описанием ссылки, указанным в правиле (т. е. описание определяет класс элементов РП).

Существующие ИС допускают следующие виды данных РП: константы; переменные, имеющие значения; сложные структуры (типа фрейм), логически объединяющие множество переменных.

В существующих ИС используются либо тривиальные виды проверок, сводящиеся к проверке наличия (отсутствия) указанных элементов в РП, либо сложные, требующие вычисления некоторых соотношений между значениями ссылок, указанных в условиях правил.

Итак, в первом приближении способ сопоставления определяется используемым типом ссылок, видом данных РП и видом проверок.

Наиболее часто в ИС используются три вида сопоставления: конкретные ссылки из правил на константные элементы в РП с проверкой их наличия; конкретные ссылки из правил на переменные в РП с вычислением соотношений; абстрактные ссылки из правил на сложные структуры в РП с вычислением соотношений. Как правило, первые два вида ссылок используются в простых ЭС, а третий — в сложных. Выполнение сложных видов сопоставления требует значительных вычислительных затрат и главным образом определяет время решения задач. Чтобы сократить его, при реализации сопоставления в ИС часто используется компиляция знаний, основное назначение которой — автоматически при создании БЗ перейти от сложного сопоставления, упрощающего описание проблемной области, к более простому, обеспечивающему эффективную работу ЭС.

Использование в решателе планирования цепочек вывода от целей предпочтительно, когда решение задач связано главным образом с определением того, какие факты необходимо знать, чтобы решить задачу, а не с желанием максимально использовать уже известные факты, в этом случае предпочтителен вывод от данных к целям.

Как правило, в ИС используются два вида объяснений: записанные объяснения и генерация объяснений по протоколу функционирования системы. В первом случае трудно добиться соответствия текста объяснений процессам, происходящим в системе, но объяснения могут содержать любую дополнительную информацию, которую эксперт считает нужным сообщить пользователю. Объяснения, полученные по протоколу работы системы, имеют то преимущество, что отражают реально происходящие в системе процессы и являются эффективным средством отладки и тестирования ЭС.

Средства приобретения знаний в существующих ИС можно оценивать с точки зрения допустимых способов формирования БЗ. Выделяют следующие способы формирования БЗ: редакторы; средства отладки; средства индуктивного вывода новых знаний. Редакторы позволяют отображать и модифицировать БЗ, возможно, в графическом виде, поддерживая ее целостность. Средства отладки обеспечивают анализ содержимого БЗ, переформирование и отображение его результатов пользователю. Средства индуктивного вывода осуществляют формирование новых знаний (правил) на основе вводимых пользователем примеров ситуаций с их решениями. Следует отметить, что средства индук-

тивного вывода и отладки БЗ в существующих ИС удается реализовать только при решении простейших задач.

В существующих ИС диалог организуется в двух видах: в фиксированной структуре и структуре, формируемой при генерации ЭС.

Среда функционирования ИС определяет тип ЭВМ, используемую операционную систему (ОС) и язык программирования. По типу используемых ЭВМ ИС можно разделить на три класса: для универсальных ЭВМ общего назначения; для персональных ЭВМ (ПЭВМ); для символьных ЭВМ и интеллектуальных рабочих станций. Выбор типа ЭВМ связан с характером проблемной области, в которой будут решаться задачи, и с основными свойствами ИС. Опыт показывает, что для работы в простых статических проблемных областях достаточно мощности ПЭВМ, для работы в сложных статических областях необходимо использовать ЭВМ общего назначения, интеллектуальные рабочие станции и символьные последовательные ЭВМ, для работы в динамических проблемных областях символьные последовательные и параллельные ЭВМ.

Язык программирования, используемый при создании ИС, позволяет судить о потенциальной эффективности и гибридности ИС. Как правило, ИС, запрограммированные на традиционных языках, более эффективны и позволяют относительно легко объединить ЭС, созданные с их помощью, с уже существующим традиционным программным обеспечением.

14.1.2. Характеристики ЭС. Перед тем как перейти к анализу конкретных ЭС, приведем обобщенную картину состояния ЭС во введенных ранее терминах (см. § 13.2).

С точки зрения назначения большинство ЭС создается для тиражирования знаний экспертов и ориентированы либо на специалистов, стремящихся повысить свои знания, либо на неспециалистов, чтобы оказать им помощь в решении сложных задач.

По характеристике «Проблемная область» ЭС обобщенно можно определить так: предметная область является статической (как правило, с неточными знаниями); решаемые задачи являются статическими (как правило, задачи расширения и реже задачи доопределения).

Приложения, реализуемые с помощью ЭС, различаются по количеству возможных решений, по сложности (структурированности) проблемной области, по методам решения задач, по типу используемых знаний. В простейших случаях, наиболее часто встречающихся на практике, существующие ЭС находят все решения в относительно малых пространствах поиска (до 10!). При этом область обычно не структурируется (описывается наборами своих характеристик), для получения решения используется полный перебор. Сложность решения подобных задач главным образом определяется отсутствием у пользователя критериев выбора решения, их неопределенностью. В настоящее время намечается тенденция создания ЭС, решающих динамические задачи в больших структурированных проблемных областях.

С точки зрения глубины анализа проблемной области (см. § 13.2) большинство существующих ЭС являются поверхностными, что обеспечивает их высокую эффективность. Однако для многих приложений необходимо создание глубинных ЭС, реализация которых требует существенных вычислительных ресурсов, что препятствует их широкому распространению.

По типу используемых методов и знаний ЭС делятся на традиционные и гибридные. Большинство существующих ЭС традиционные. Тенденцией является создание гибридных ЭС путем выделения в неформализованной задаче формализуемых подзадач и реализации их методами традиционного программирования (т. е. в виде БД и ППП). Гибридизация ЭС значительно усложняет процесс управления системой, увеличивает количество неявно представленных знаний, что ухудшает ее объяснительные возможности и сужает номенклатуру используемых методов инженерии знаний.

С точки зрения класса ЭС большинство существующих ЭС являются простыми. Так, в 1985 г. в США было продано 12 тыс. ЭС, из них 11 тыс. простых.

При анализе состояния ЭС основное внимание будет уделено ЭС, достигшим промышленной или коммерческой стадии, так как только об этих ЭС можно говорить, что они используются при решении практических задач.

С точки зрения ИС большинство ЭС на ранних стадиях существования (см. § 13.2) используют оболочки и средства автоматизации проектирования. На промышленных и коммерческих стадиях используются либо символьные языки (особенно при работе на символьных ЭВМ), либо традиционные языки программирования.

В соответствии со сказанным большинство существующих ЭС могут быть отнесены к первому поколению, так как они являются статическими и поверхностными. В настоящее время появляются ЭС второго поколения и начинаются работы по реализации ЭС третьего поколения.

14.2. Состояние разработки инструментальных средств

Здесь будут рассмотрены наиболее популярные ИС, достигшие коммерческой стадии развития. Поскольку символьные языки программирования и языки инженерии знаний подробно рассматриваются в кн. 3 данного Справочника, то здесь акцент будет сделан на рассмотрении ИС, являющихся оболочками и средствами автоматизации проектирования (разработки). Рассмотрение ИС будет осуществляться по типам (оболочки ЭС, средства автоматизации проектирования ЭС, языки инженерии знаний). Сводный перечень наиболее важных ИС приведен в табл. 14.1.

14.2.1. Оболочки ЭС. В качестве типичных ИС, относящихся к оболочкам, можно назвать такие, как EXSYS, SAGE, SAVOIR, EMYCIN. Основные свойства оболочек перечислены в п. 14.1.1. Рассмотрим теперь их более детально на примере системы SAVOIR фирмы ISI Limited.

Система SAVOIR содержит все основные компоненты ЭС (см. § 13.1) и рассчитана на создание ЭС пользователями, не являющимися специалистами в области программирования. Система работает в режиме приобретения знаний и режиме консультации. Каждый режим представляется отдельным программным компонентом. Пользователям может предоставляться только компонент ведения консультации. Система разработана для следующих ЭВМ: IBM, IBM PC(XT, AT), VAX, APRICOT, SIRIUS, PRIME — и работает с различными ОС. Управление решением задач включает возможность построения вывода от целей и от данных.

Для представления фактов о проблемной области в ЭС используются переменные следующих типов: 1) вероятностный — для представления информации, которая может быть неточна; 2) логический — для представления информации, которая может принимать значения истина-ложь; 3) числовой — для представления количественной информации; 4) строчный — для хранения текстов; 5) вопрос — значение которого устанавливается путем задания вопроса пользователю. В системе определены следующие типы вопросов: вопрос для выбора значения из списка значений; вопрос для получения числа; вопрос для получения строки; вопросы, требующие ответа да или нет.

Решение задачи осуществляется на основе сети вывода, которая строится компилятором системы в соответствии со знаниями, полученными от пользователя в виде правил. Сеть вывода в явном виде содержит связи между переменными и условия, при которых возможны вычисления значений одних переменных через другие.

Система обеспечивает работу с недостоверными знаниями и данными. Оценка правдоподобности знаний осуществляется с помощью коэффициентов определенности. Манипулирование коэффициентами определенности может осуществляться либо байесовским методом, учитывающим вклад в результат всех наблюдений, либо на основе операций нечетной логики, позволяющей работать с расширениями функций И, ИЛИ, НЕ, оперирующими с недостоверными данными. Система позволяет через функции CALL вызывать внешние функции, реализованные на традиционных языках программирования.

Взаимодействие с пользователем в режиме как приобретения знаний, так и консультации осуществляется с помощью цветного многооконного интерфейса. Средства объяснения обеспечивают типичные для ЭС виды объяснений и работу системы в режиме трассировки, когда пошагово объясняются все действия системы. Система является типичной оболочкой, обеспечивающей минимум затрат для создания прототипов и ЭС.

В системе SAVOIR используется упрощенный набор средств представления знаний и процедур вывода. Это позволило реализовать систему SAVOIR на ПЭВМ и мини-ЭВМ. Пользователь обучается работе со средствами самостоятельно в ходе работы с системой или на основе введенных в систему уроков. Следует отметить, что оболочки типа SAVOIR не позволяют строить ЭС, эффективно работающие в больших и сложных проблемных областях.

Итак, SAVOIR является типичным примером оболочки. Существуют более сложные оболочки, например SI, PERSONAL CONSULTANT+. Усложнения связаны главным образом с вводом средств представления знаний о структуре области, реализацией более сложных механизмов сопоставления, с введением механизмов наследования свойств, использованием нескольких стратегий решения задач и введением средств, допускающих выбор желаемой стратегии.

Широкое распространение получили более простые ИС (например, EXPERT-EASY), которые не включают работу с неточными знаниями и ограничиваются представлением в предметной области только наборов утверждений. Однако простота этих оболочек позволяет автоматизировать процесс приобретения знаний за счет средств индуктивного вывода, что существенно упрощает приобретение правил. Обычно данные средства обрабатывают примеры ситуаций, сообщаемых пользователем, и формируют на их основе правила вывода.

14.2.2. Средства автоматизации проектирования ЭС. Типичными средствами автоматизации проектирования ЭС в настоящее время являются такие системы, как KEE, ART, ACORN, Knowledge Craft.

Рассмотрим более конкретно особенности этого типа ИС на примере систем KEE и ACORN.

Система KEE фирмы Intellicorp является гибридным универсальным ИС, объединяющим несколько парадигм искусственного интеллекта. Она обеспечивает работу: с системой представления знаний, основанной на фреймах; механизмом наследования свойств; логическим представлением; выводом, основанным на правилах; выводом, управляемым данными; объектно-ориентированным программированием; механизмом поддержания альтернативных миров и системой поддержания истинности.

Система KEE функционирует на таких типах ЭВМ, как LMI Lambda, Hexox 1100, Symbolics, TI Explorer, Sun-3, Dec Vax, HP 9000.

Основу представления знаний о проблемной области в системе KEE составляют фреймы, определяемые именем и описаниями атрибутов. Описания атрибутов включают: ограничения на значения атрибута; процедуры, которые должны вызываться, когда изменяется или устанавливается значение атрибута; описание источников и наследование значения атрибута. Значения атрибутов могут либо быть изначально установлены, либо наследоваться из других фреймов, либо вычисляться процедурами на языке LISP, либо выводиться с помощью правил. Различаются прототипы фреймов (определяющие, какие сущности могут присутствовать в проблемной области) и их реализации, представляющие сущности, имеющие место в конкретной ситуации, в рамках которой осуществляется решение задачи. Фреймы объединяются в иерархическую сеть наследования свойств с исключениями.

Для представления процедурных знаний в системе KEE кроме процедур используются правила. Правила могут включать как логические выражения с конъюнкцией, дизъюнкцией и отрицанием, так и произвольные LISP-выражения. Ссылки из правил абстрактны.

Интерпретатор правил обеспечивает прямой или обратный вывод со стратегией перебора в ширину или в глубину и механизм разрешения конфликтов в

Наименование	Стадия существования	Тип	Представление знаний	
ACLS	Коммерческая	Оболочка	Примеры, правила	
ADVISE	Исследовательский прототип	Средство автоматизации проектирования ЭС	Правила	
AGE	Исследовательская	То же	.	
Analyser Plus	Коммерческая	.	.	
APES	Исследовательская	Язык инженерии знаний		
APES	Коммерческая	Средство автоматизации проектирования ЭС		
APLICOT	Экспериментальная	Язык инженерии знаний	Логика	
ART	Коммерческая	Средство автоматизации проектирования ЭС	Правила, прямые и обратные цепочки	
Autologie	.	То же	Правила, прямые цепочки	
Cash Value	.	Язык инженерии знаний	Правила	
CODES	.	То же	Фреймы	
Cristal	.	Оболочка	Правила, обратные цепочки	

Основные средства

Основные свойства	Среда функционирования	Источник
Обеспечивает преобразование внешнего представления правил, получаемых при решении экспертом задачи, в общие правила методом индукции	APPLE, IBM PC; PASCAL	Paterson et al., 1982
Вспомогательные программы для различных форм представления знаний, стратегий управления, индуктивного вывода правил и т. д.	PASCAL	Michalski et al., 1983
Выбор фреймов, языка, правил, соединение отдельных компонентов в систему	INTERLISP	Aiello et al., 1981; Nii et al., 1979
Предоставление возможности экспертам разрабатывать ЭС без участия инженеров по знаниям	IBM PC	Expert Systems, 1983, v. 3, N 1
Обработка недостоверной информации	IBM PC (128K): PROLOG	Hammond, 1982; Hammond, 1983
Автоматическая генерация диалога и текстов объявлений	IBM PC (128K)	Expert Systems, 1985, v. 2, N 4
Гибкое управление, использующее прямые и обратные цепочки; обработка фактора уверенности. Вспомогательная среда; интерактивный редактор правил и фраз	ICOT, DEC-2060	Mizoguchi, 1983
Сравнение альтернативных решений и выбор оптимального с учетом «точки зрения»; решение задач логического программирования, составление планов и графиков (см. § 14.2)	VAX, Symbolics, SUN-8, Explorer, Xerox 1100; LISP	Expert Systems, 1985, v. 2, N 4
Вывод правила на основе примеров; несколько редактирующих программ; объяснительная способность	IBM PC	Expert Systems 1985, v. 2, N 4
	IBM PC (192K)	To же
Задачи диагностики	IBM PC (256K)	»
Для введения правил используется интерактивный редактор; нет необходимости овладевать языком правил	IBM PC	Expert Systems, 1986, v. 3, N 3; Expert Systems User, 1987, v. 2, N 12

Наименование	Стадия существования	Тип	Представление знаний
DPL	Исследовательская	Язык инженерии знаний	Фреймы
E Fossil	Коммерческая	Оболочка	Деревья
ESDE ESCE	"	"	Прямые и обратные цепочки, правила
ES/P Advisor	"	"	Правила
ESP Frame Engine	"	"	Прямые и обратные цепочки, демоны, механизм наследования; фреймы
ETS	Исследовательская	Средства автоматизации проектирования ЭС	
Exper-teach-II	Коммерческая	То же	Правила
Expert-4	"	Оболочка	Правила, размытая логика
Expert	Исследовательская	"	Правила
EXPERT-EASE	Коммерческая	"	Примеры, правила
Expertkit	"	"	Продукционные правила, прямые цепочки

Основные свойства	Среда функционирования	Источник
Создание и манипулирование фреймами. Интерактивный графический редактор Возможности для БЗ: режимы запроса эксперта и обучения. Объяснительные способности, графика, ЕЯ-интерфейс. Для областей с таксономическим представлением знаний (геология и т. п.) Для решения задач «структурного выбора» (диагностики); не требуется инженера по знаниям в процессе работы; встроенная объясняющая программа Преобразование заранее подготовленных текстов в ЭС, до 400 правил Для опытных разработчиков ЭС в области диагностики, конфигурации, планирования Помощь экспертам в создании и анализе БЗ путем интервьюирования эксперта Комплексный инструментарий, включающий примеры ЭС, интерактивные обучающие программы, языки ИИ и 6 оболочек ЭС, работающих с правилами Для студентов и специалистов в области биомедицинской информации (классификация и диагностика) Для решения задачи диагностики и классификации. Широко используются в медицине Помощь при создании ЭС; требует от эксперта примеры условий, ведущих к каждому результату, и создает дерево вывода Диагностика, управление обработкой данных в реальном масштабе времени. До 1000 правил	Symbolics-3600, ZETALISP PROLOG IBM System/370; VM/CM; PASCAL IBM PC; PROLOG IBM PC/AC (640K) Xerox-1100; INTERLISP-D IBM PC (256K) LISP, PROLOG, PASCAL IBM PC (256K); Compiled, BASIC DEC IBM; FORTRAN IBM PC/ XT (128K); PASCAL IBM PC, Macintosh; Le-LISP	Batali, 1980; Davis et al., 1982 Expert Systems, 1986, v. 3, N 2 Expert Systems, 1985, v. 2, N 4 Expert Systems, 1985, v. 2, N 4; Expert Systems User, 1987, v. 2, N 12 Expert Systems User, 1987, v. 2, N 12 Boose, 1984 Byte, 1987 Febr. Expert Systems User, 1987, v. 2, N 12 Waterman, 1986 Perrone, 1983 Expert Systems User, 1987, v. 2, N 12

Наименование	Стадия существования	Тип	Представление знаний
Exsys	Коммерческая	Оболочка	Правила
Ex-TRAN-7 (Expert translator)	"	"	Правила, примеры
First-Class	"	"	Индукции; правила
FIT	Исследовательская	Язык инженерии знаний	Логика
G2	Коммерческая	Средство автоматизации проектирования	Правила, фреймы
CETREE	Исследовательская	Средство автоматизации проектирования ЭС	Сети, правила
GURU	Коммерческая	То же	Правила
Gold Hill Expert System Toolkit (-Acorn Shell)	"	"	Правила, фреймы
GPSI	Экспериментальная	"	Правила
IKE	Коммерческая	Средство автоматизации проектирования	Правила, обратные цепочки, деревья
IN-ATE/KE	"	Оболочка	
Insight-I Insight-2	"	"	Продукционные правила, прямые цепочки
Intelligence Service (ISExpert Consultant)	"	"	Прямые и обратные цепочки, продукционные правила

Основные свойства	Среда функционирования	Источник
Обработка вероятности; может служить интерфейсом систем БД, вызывать внешние управляющие программы Не требует знания программирования, включает пакет вывода правил из примеров, интерактивные возможности, благодаря которым система может использоваться как самоучитель Средство для разработки ЭС, не требующее от пользователя знания программирования Объединение свойства Пролога и ЛИСП. Широкая ориентация, нехронологический параллелизм Создание систем для управления процессами в реальном времени Вспомогательное средство для управления БЗ в виде сетей; представление правил в виде графа И/ИЛИ ЕЯ-интерфейс для расчетов; интерфейс с БД, электронная таблица См. п. 14.2.3 Вспомогательная программа для создания систем, основанных на правилах Меню; графический интерфейс, представляющий в виде деревьев правила; обратные цепочки учитывают фактор уверенности Вспомогательная программа для создания диагностирующих ЭС Два модуля: язык продукционных правил и компилятор; система вывода, формирующая БЗ Для неподготовленного пользователя и эксперта. Диагностика, принятие решений, до 300 правил	IBM PC; C UNIX, VAX/VMS, IBM PC; FORTRAN-77 IBM PC (PC DOS, MS DOS, 256K) DEC-10; UCI-LISP Symbolics, Common-LISP DEC-VAX, VMS IBM PC, VAX 11/780, PC WOS, MS DOS; C IBM PC; GCLISP Lambda-LISP IBM PC; TURBO-PASCAL IBM PC (MS, DOS, 512K); TURBO-PASCAL	Expert Systems, 1985, v. 2, N 4 Expert Systems, 1985, v. 2, N 2; Expert Systems User, 1987, v. 2, N 12 Expert Systems, 1986, v. 3, N 3 Boley, 1983 Вулф, 1987 Lewis et al., 1983 AI, 1987a Expert Systems User, 1987, v. 2, N 11 Harandi et al., 1983, 1984 Expert Systems, 1985, v. 2, N 4 Cantone et al., 1983 То же Expert Systems User, 1987, v. 2, N 12

Наименование	Стадия существования	Тип	Представление знаний
KANDOR	Исследовательская	Язык инженерии знаний	Фреймы
KAS	Коммерческая	Оболочка	Правила
KEE	То же	Средство автоматизации проектирования	Правила, фреймы, логика
KES	•	То же	Схемы, ассоциации, правила
KISS	•	•	Обратные цепочки, индукция
KL-ONE	Исследовательский прототип	Язык инженерии знаний	Семантические сети
KMS	То же	Средство автоматизации проектирования ЭС	Правила, фреймы
Knowledge Craft	Коммерческая	То же	Правила, фреймы, Логика
Micro-IN-ATE	•	Оболочка	Правила

Основные свойства	Среда функционирования	Источник
Преднамеренная ограниченность выразительности языка для обеспечения простоты вычислений; наследование свойств, простой интерфейс Интерфейс и механизм вывода от системы PROSPECTOR См. § 14.2.2	LISP INTERLISP VAX, Xerox 1100; TI Explorer, Symbolics	Patee Scheider, 1984 Expert Systems, 1986, v. 3, N 1 Kehler et al., 1984; Kunz et al., 1984
Представляет собой соединение трех частей, каждая из которых имеет свою схему представления знаний и механизм вывода: 1) системе продукционных правил; 2) систему статических моделей; 3) систему гипотез и тестов Поиск, хранение и вывод знаний	Cyber 180, VAX, APOLLO, IBM PC и др. C, IQLISP IBM PC/XT, AT и др. (DOS); Micro-Prolog Professional	Expert Systems, 1985, v. 2, N 4 Expert Systems User, 1987, v. 2, N 12 Schmolze et al., 1982;
Автоматическое наследование свойств, автоматический классификатор. Интерактивный графический редактор БЗ Множество подсистем, каждая имеет свое представление знаний и свои методы вывода. Дедукция, основанная на правилах, статистические методы классификации, порождение выводов, основанных на фреймах	Xerox 1100, DEC VAX; Symbolics, TI Explorer; INTERLISP Univac 1100/40; LISP	Schmolze et al., 1983 Reggia, 1981
Программное обеспечение для создания больших систем, основанных на знаниях (см. § 14.2)	Symbolics T, VAX, TI Explorer; OPS-5, PROLOG	Expert Systems, 1986, v. 3, N 1
Приобретает информацию, необходимую для формирования БЗ из данных средств автоматизации проектирования и данных о достоверности	APPLE, Macintosh, IBM PC; PASCAL, LISP	Expert Systems, 1986, v. 3, № 1

Наименование	Стадия существования	Тип	Представление знаний
MORE	Исследовательская	Средство автоматизации проектирования ЭС	Правила
LEONARDO	Коммерческая	СБОЛОЧКА	.
LOOPS	Исследовательская	Объектно-ориентированный язык	Правила, процедуры, демоны
OPS-83	.	Язык инженерии знаний	Правила, прямые цепочки
OPS-5+	.	То же	То же
OPS-5—	.	.	.
OPS-5e	.	.	.
NEXUS	Коммерческая	Оболочка	Правила, прямые и обратные цепочки
PC/Beogle	.	.	Прямые цепочки, булевы правила, таблица вероятностей
Personal Consultant Plus	.	.	Правила, фреймы
PICON	.	Средство автоматизации проектирования	Правила
Pro MD	.	Оболочка	
PLL	Исследовательский прототип	Язык инженерии знаний	Фреймы

Основные свойства	Среда функционирования	Источник
Порождает правила диагностики на основании интервью с экспертами; строит модель соответствующей области; используется фактор уверенности Имеет средства для неподготовленных пользователей и для инженеров по знаниям. Объективно-ориентированная архитектура Включает четыре парадигмы программирования. Вспомогательная система, включающая ИС для отладки. См. п. 14.2.2 Версия языка OPS-5, в которой механизмы представления знаний сочетаются с обычными программными структурами Расширение системы OPS-5 (меню, «окна», «мышь») Дешевая версия языка OPS-5 Расширенная версия языка OPS-5 для LISP-машин Финансовый анализ, экономика, управление, коммерция. 500 производционных правил Вывод правил из данных для пользователя и для инженера по знаниям 2000 правил Для управления производственными процессами в реальном масштабе времени Задачи диагностики в медицине. В основу MYCIN; ЕЯ-интерфейс Благодаря гибкости возможно уточнение языка представления с помощью частных представлений, стратегий наследования и схемы управления	IBM XT, AC, PC DOS; FORTRAN, ASSEMBLER Xerox-1100; INTERLISP-D IBM PC, DEC VAX, APOLLO, SUN APOLLO, IBM PC, Macintosh; C Unix, CP/M, MS DOS Symbolics 3600, TI Explorer IBM PC/XT, AT; FORTH IBM PC, DEC VA, Data General MV 11p300; TURBO-PASCAL, C IBM PC, TI Explorer (512K) Lambda LMI; LISP IBM PC; PROLOG-2 INTERLISP	Kahn et al., 1984в Expert Systems User, 1987, v. 2, N 12 Waterman, 1986 Expert Systems User, 1987, v. 2, N 10; Expert Systems, 1985, v. 2, N 4 Expert Systems, 1985, v. 2, N 4 To же Expert Systems User, 1987, v. 2, N 10 Expert Systems User, 1987, v. 2, N 12 Expert Systems User, 1987, v. 2, N 12 Watterman, 1986 Moore et al., 1984; Morris, 1984 Expert Systems User, 1987, v. 2, N 12 Greiner et al., 1980; Watterman et al., 1983

Наименование	Стадия существования	Тип	Представление знаний
RULE MASTER (версия 3.0)	Коммерческая	Средство автоматизации проектирования	Правила
RULE MASTER	"	То же	"
S-1	"	Оболочка	Правила, фреймы
SAVOIR	"	"	Правила
SRL +	"	Язык инженерии знаний	Фреймы, правила
SRL/1.5	Исследовательский прототип	То же	То же
Super Expert	Коммерческая	Средство автоматизации проектирования ЭС	Правила
Smalltalk-AT	"	Сбъектно-ориентированный язык	Объекты
Smalltalk-V	"	То же	"
Smalltalk-80	"	"	"
TEIRESIAS	Исследовательский прототип	Средство автоматизации проектирования ЭС	Правила

Основные свойства	Среда функционирования	Источник
Автоматическая индукция правил из примеров; объяснения (почему?), использование «размытой» логики Вывод правил на основании примеров, иерархия порождающих правил, возможность использования внешних данных и процессов, автоматическая генерация объяснений Взаимодействие с другими программами, простота в обращении, не требует переучивания программистов Допускает правдоподобные рассуждения, контролирует наполнение БЗ; визуальное представление данных. См. п. 14.2.1 Для логических, объектных и ориентированных на правила методов представления. Наследование свойств определяется пользователем; встроенная система ведения БД Автоматическое и заданное пользователем наследование свойств, разнообразные контексты для суждений в альтернативных мирах. Каждому фрейму могут приписываться метазнания «Индуктивное» средство разработки ЭС, выводящее правила из серии примеров (до 1000 примеров) Версия языка Smalltalk с графическим интерфейсом, «мышью», «окнами». Совместим с Smalltalk-80 Версия языка Smalltalk с организацией памяти в двоичных кодах Расширение стандартной версии языка Smalltalk Передача знаний эксперта в БЗ в интерактивном режиме. Ограниченный подязык английского языка для приобретения новых правил	MS DOS IBM PC/ AT, XT; C VAX, SUN, IBM PC; C SUN, APOLLO, Micro-VAX, Xerox 1100; Symbolics; C, ADA IBM 370, IBM PC/XT; PRIME, SYRIUS, VAX DEC VAX, Symbolics-3600; COMMON-LISP DEC VAX; FRANZ-LISP. IBM PC/ XT, AT IBM PC IBM PC Textronics-4400 INTERLISP	Byte, 1987, Jan. Michie et al., 1984; Riese, 1985 Expert Systems, 1985, v. 2, N 4 Expert Systems, 1985, v. 2, N 4 Expert Systems User, 1987, v. 2, N 12 Wright et al., 1983 То же Expert Systems User, 1987, v. 2, N 4 Expert Systems, 1987, v. 2, N 10 То же » Davis, 1976; 1978; 1979

Наименование	Стадия существования	Тип	Представление знаний
TESS	Коммерческая	Оболочка	Правила на ограниченном английском языке
TIMM	•	Средство автоматизации проектирования ЭС	Правила
VP-Expert	•	То же	•
Twaice	•	•	Правила, прямые и обратные цепочки
WIZDOW	•	Оболочка	Семантические сети
WPS	•	•	Правила
Xi	•	•	Прямые и обратные цепочки
Xi Plus	•	•	Прямые и обратные цепочки, правила
XSYS	•	•	Правила
ДИРП	Промышленная	•	Правила сети, процедуры
ДИЭС	Коммерческая	•	Правила, утверждения, числовые переменные. Компиляция правил в сеть

Основные свойства	Среда функционирования	Источник
Для разработки небольших ЭС и экспериментирования в области технологии. ЕЯ-интерфейс, интерактивные возможности. Работает с неточными знаниями	IBM PC (256K); C	Expert Systems, 1985, v. 2, N 2
Эксперт составляет список всех возможных решений, факторов и их оценок; система на основе примеров выводит правила типа «Если — то».	VAX 11/780, IBM PC и др. (512K); FORTRAN-77	Kiselewich, 1983; Kornell, 1984, Parker et al., 1984
Для обработки вероятностной информации используется фактор уверенности		
Для создания ЭС. Вывод правил из примеров	IBM PC	Byte, 1987, June
Правила запрашиваются интерактивным редактором и затем составляются для данной версии ЭС. Проверка непротиворечивости правил	NIXDORF-8832; PROLOG	Expert Systems, 1986, v. 3, N 1
Оболочка для создания и отладки ЭС	IBM PC/AT (640K); C	Expert Systems, 1986, v. 3, N 3
Для решения задач планирования, оценки производительности	IBM PC/XT, AT; C	Expert Systems User, 1987, v. 2, N 12
Возможны вопросы типа «что, если ...»	IBM PC; PROLOG	Expert Systems, 1985, v. 2, N 4
Для программистов, экспертов и пользователей в области законодательства и диагностики. Динамическое распределение памяти	IBM PC/XT, AT; MicroPROLOG, ASSEMBLER	Expert Systems User, 1987, v. 2, N 12
Обработка приближенной информации	IBM PC; LISP	Expert Systems, 1985, v. 2, N 4
Представляет причинные, временные и пространственные знания	ЕС ЭВМ; ПЛ/1	Кириллов и др., 1985
Использует байесовские, логические и арифметические правила (до 1000 правил). Работает с неточными данными и знаниями	ЕС-1840, ЕС-1841, АЛЬФА ДОС-16, IBM PC/XT, AT, MS DOS	Кирсанов и др., 1989

Наименование	Стадия существования	Тип	Представление знаний
ЭКО	Коммерческая	Оболочка	Правила, утверждения, числовые переменные Компиляция правил в сеть
ИДИС	Исследовательская	"	Правила
КОНС-ПРОЛОГ	"	"	"
НЭКС-2	Коммерческая	"	Объекты, правила, метаправила, функции
МикроПРИЗ МикроЭКСПЕРТ	"	Средство автоматизации проектирования	Правила, сети
ПИЭС	Исследовательская	То же	Правила
Реляп	"	Язык инженерии знаний	Отношения, правила
СПЭИС	"	Оболочка	Фреймы, правила
ФИАКР	Коммерческая	"	Правила
ЭКСПЕРТ	"	"	"
ЭКСПЕРТИЗА (ГЛОБ)	"	"	Фреймы, правила

случаях, когда возможно применение нескольких правил одновременно. Итеративные вычисления значений атрибутов в системе не поддерживаются.

Механизм альтернативных миров обеспечивает поддержание в системе различных точек зрения на текущую ситуацию в проблемной области в соответствии либо с ответами пользователя, либо с предположениями, сделанными в ходе вывода. Система поддержания истинности обеспечивает автоматический пересмотр всех выводов, сделанных из предположения, которое оказалось некорректным.

Система КЕЕ имеет развитые встроенные графические средства объяснения и тестирования как динамических процессов, происходящих в ходе решения задачи, так и статического состояния БЗ. В качестве средств приобретения знаний используется интеллектуальный редактор БЗ. Система КЕЕ является эф-

Основные свойства	Среда функционирования	Источник
Развитие системы ДИЭС. Осуществляет взаимосвязь с внешними программами. Использует лингвистические шкалы и изображения при организации диалога. Позволяет строить иерархические ЭС, объем базы знаний Для систем диагностики технических объектов Для задач диагностики небольшого размера Несколько стратегий вывода, метапланирование, работа с альтернативными мирами и предположениями, связь с СУБД Для построения гибридных систем на основе синтеза программ и эвристических знаний Многоуровневая с несколькими языками представления: ПИЛОТ, ATNL2, TUTOR Реляционный язык представления знаний Для задач диагностики То же » Для задач диагностики, гибридная система	ЕС-1841, Альфа-ДОС-16, IBM PC/AT, XT, MS DOS, Си IBM, PC, MS DOS; C СМ-4, ОС РАФОС-2; Пролог ЕС-1060, ОС 6.1, 70, IBM PC/AT с дополнительной 4-Мбайт памятью и процессором 80386 ЕС-1840, IBM PC/XT, Labtam IBM PC/XT, AT, Labtam 3000; Рефал/2 ЕС ЭВМ IBM PC, MS WOS; Лисп СМ-4; Паскаль IBM PC, ЕС-1840, ЕС-1841, Альфа-ДОС; Си ЕС ЭВМ, ОС 6.1, 7.0; ЛИСП ВМ, Ассемблер ГЛОБ на ПЭВМ	Кирсанов и др., 1989 Голубев и др., 1987 Диалоговая, 1987 Ивашенко и др., 1987 Коов, 1987 Хорошевский, 1986 Kleshchev, 1984 § 15.1 § 15.3 § 15.2 Гл. 16

фективным средством автоматизации проектирования ЭС, ориентированным на использование в различных приложениях и обеспечивающим настройку на их особенности. Однако стоимость этого программного продукта сопоставима со стоимостью техники, на которой он эксплуатируется, а эффективное его использование предполагает дорогостоящее обучение пользователей.

Система ACORN фирмы Gold Hill Computers является универсальным средством, которое интегрирует в себе фреймы с наследованием свойств, построение прямых и обратных цепочек вывода, объектное программирование и присоединенные процедуры. Многоуровневая архитектура системы ACORN позволяет настраиваться на особенности конкретного приложения. На верхнем уровне ACORN является оболочкой ЭС, позволяющей создавать и отлаживать ЭС. На среднем уровне ACORN организует высокоуровневый доступ к своим

компонентам: языку представления знаний, механизму наследования свойств, системе поддержания истинности, средствам манипулирования коэффициентами определенности, средствам объяснения и интерфейсу с внешними программными средствами. На нижнем уровне ACORN предоставляет средства LISP-системы с расширенной памятью, макросредствами и средствами отладки.

Система ACORN работает на ПЭВМ IBM PC AT, но полные его возможности можно использовать при применении микропроцессоров Intel 80286 и 80386, работающих с памятью в 15 и 24 Мбайт, или специализированных LISP-машин. В ходе решения задач в ACORN выполняется сопоставление сложных структур с использованием наследования свойств между структурами (фреймами). Системы ART и Knowledge Craft во многом аналогичны рассмотренным системам. Системы KEE и Knowledge Craft являются расширением ранее существовавших фреймовых систем со свойственными таким системам механизмами присоединенных процедур, ограничениями на значения атрибутов и т. д. Система ART является развитием системы, основанной на правилах, и в ней отсутствует ряд упомянутых свойств фреймовых систем, но более продвинуты свойства, связанные с управлением правил. Вводятся различные типы правил: гипотезирования, ограничения, правдоподобия. Различаются правила вывода и продукционные правила, позволяющие изменять или итеративно вычислять значения атрибутов.

14.2.3. Языки инженерии знаний. Рассмотрим языки инженерии знаний на примере объектно-ориентированного языка LOOPS и продукционного языка OPS.

В языке LOOPS все понятия и процессы представляются в виде сложных структур-объектов, снабженных своими средствами вычислений — методами. Процесс решения задач сводится к посылке и выполнению сообщений объектами. Объекты множественным образом наследуют друг у друга свойства (временные и методы). Кроме объектного программирования язык LOOPS обеспечивает возможность задания процессов в виде правил, демонов, активных значений, обладает развитыми графическими средствами взаимодействия с пользователем и отображения процессов, происходящих в ходе решения задач. Весь комплекс средств языка LOOPS позволяет достаточно легко программировать компоненты ЭС.

Язык OPS имеет множество реализаций (OPS-4, OPS-5, OPS-83 и т. д.). Основу всех реализаций составляет: 1) описание состояний предметной области в виде переменного числа изолированных объектов, имеющих имя и совокупность поименованных атрибутов со значениями; 2) описание процессов в виде правил, проверяющих наличие в рабочей памяти требуемых объектов, удовлетворяющих заданным условиям и модифицирующих содержимое рабочей памяти; 3) быстрый механизм сложного сопоставления, основанный на RETE-алгоритме [Forgy, 1982]. Язык OPS не является завершенным инструментом для создания ЭС, в нем отсутствуют средства объяснения и приобретения знаний, весьма ограничены средства тестирования, он труден для использования непрограммистами. Однако применение при сопоставлении RETE-алгоритма делает его одним из наиболее быстрых ИС. Имеются реализации языка на различных типах ЭВМ, включая ПЭВМ.

14.3. Состояние разработки экспертных систем

Здесь рассмотрены наиболее популярные ЭС. Предпочтение отдано тем ЭС, опыт разработки которых оказал заметное влияние на создание других ЭС или ИС. В табл. 4.2 приведены наиболее важные экспертные системы. Некоторые разделы таблицы не заполнены либо в силу отсутствия сведений, либо в силу того, что их содержимое очевидно вытекает из содержимого других граф.

Рассмотрение систем осуществляется в соответствии с типом задач, решаемых ЭС (расширение, доопределение, преобразование; см. § 13.2). Поскольку

основным классом решаемых в настоящее время задач являются задачи расширения, то им уделено наибольшее внимание.

14.3.1. ЭС, решающие задачи расширения. Среди систем этого класса наиболее известными являются системы PROSPECTOR, MYCIN и R1. Методы решения задач, разработанные при создании системы PROSPECTOR, легли в основу таких ИС, как KAS, SAGE, MICRO-EXPERT, SAVOIR. На основе системы MYCIN созданы ИС EMYCIN, S1, серия ЭС PUFF, SACON, VM, LIHO, ONCOCIN. Система R1 (XCON) продемонстрировала возможности чисто продукционного языка OPS для построения ЭС, что привело к созданию на его основе целой серии как ЭС (XSEL, IDI, YES/MVS, MUD), так и ИС, использующих для представления знаний правила OPS.

Несмотря на относительную простоту перечисленные выше системы достаточно тщательно реализованы и полное описание заложенных в них механизмов требует значительного объема. Нас будут интересовать в первую очередь те детали этих систем, которые были использованы в последующих разработках.

Система PROSPECTOR предназначена для выдачи геологам следующей информации: о наличии в анализируемой местности залежей информации; о наличии в анализируемой местности залежей полезных ископаемых; об оценке геологических ресурсов района; о выборе мест, благоприятных для бурения. Система создана фирмой SRI International. Работы над системой начаты в 1974 г. и продолжались до 1983 г. Трудозатраты на создание системы оцениваются в 30 чел.-лет, система достигла промышленной стадии. При этом БЗ системы содержит свыше 1000 правил и включает классификацию более 1000 геологических понятий.

Система PROSPECTOR решает задачи чисто поверхностным образом, не вникая в суть происходящих в проблемной области процессов. Состояние проблемной области описывается в виде утверждений о проблемной области, например «условия района благоприятны». Решение задач состоит в изменении априорно установленных оценок «вероятностей» утверждений, которое осуществляется по результатам наблюдений, сообщаемых системе пользователем. Пересчет оценок происходит на основе формулы Байеса, связывающей апостериорную вероятность гипотезы с ее априорной вероятностью и вероятностью наблюдения. Взаимосвязь гипотез и наблюдений представлена в виде сети, которую можно рассматривать как графическое представление правил. Для представления правил в сети имеются И и ИЛИ вершины, описывающие условия правил. Дополнительно в системе для выражения таксономии геологических понятий используется семантическая сеть. Представление таксономии понятий позволяет системе делать некоторые выводы, например устанавливать, что наличие пиритов означает и наличие сульфидов. Основной управляющей стратегией системы является поиск от целей вглубь с применением ряда эвристик, обеспечивающих максимальное повышение шансов гипотез. В системе реализованы объяснительные возможности, выявляющие причины задания вопроса или состояния решения задачи. Во время решения задач не вводятся новых утверждений о проблемной области. Все утверждения должны быть априорно перечислены экспертом.

Инструментальные средства, в основу которых положена система PROSPECTOR, обеспечивают удобные средства решения задач в областях, характеризующихся неполнотой и неточностью знаний и возможностью описать состояние области с достаточной для решения задач степенью подробности в виде заранее фиксированного набора утверждений.

Система MYCIN решает примерно те же задачи, что и система PROSPECTOR, но делает это несколько иначе и не требует фиксированного набора утверждений о проблемной области. Система разработана в Стэнфордском университете и достигла стадии действующего прототипа. Назначение системы — оказание помощи лечащим врачам в постановке диагноза и назначении лечения в сложных случаях инфекционных заболеваний крови. Цель системы — опреде-

Название	Область применения	Назначение	Стадия существования
ABEL	Медицина	Анализ нарушений кислотно-щелочного баланса	Исследовательский прототип
ACE	Электроника	Обнаружение и устранение неисправностей телефонной сети	Коммерческая
ACES	Военные приложения	Выполнение картографических работ	Исследовательский прототип
Achilles	Промышленность	Анализ проблем коррозии и выдача рекомендаций по отбору материалов для применения в корродирующей среде	Промышленная
AIDA	Вычислительная техника	Анализ дампа, типов задач, эргономическое представление данных из дампа	Исследовательский прототип
AI/MM	Медицина	Анализ деятельности почек	Демонстрационный прототип
AI/PHEUM	.	Анализ заболеваний соединительных тканей в клинической ревматологии	Исследовательский прототип
AIRID	Военные приложения	Идентификация самолетов на основании визуальных наблюдений	Демонстрационный прототип
AIRPLAN	То же	Помощь при запуске самолетов, предупреждение аварийных ситуаций	Ограниченно действующий прототип
AI-SPEAR	Вычислительная техника	Выявление ошибок на магнитной ленте и предложения по их исправлению	Действующий прототип
AMUID	Военные приложения	Анализ поля боя, классификация целей в реальном масштабе времени	Демонстрационный прототип
ANALIST	То же	Обнаружение и классификация целей	То же
ANGY	Медицина	Анализ заболеваний коронарной артерии	»
APEX	Финансовое дело	Помощь финансистам при планировании и управлении вкладами клиентов	Прототип
ATR	То же	Определение и классификация цели по движущимся образцам	Демонстрационный прототип

системы

Представление знаний	Тип задачи	Инструментарий	Источник
Семантические сети	Диагностика		Paterson et al., 1982
Правила, прямые цепочки	Наладка, диагностика	OPS-4	Vesonder, 1983
Объектно-ориентированное		LOOPS	Waterman, 1986
	Диагностика		Simmonds, 1985
		LISP	Hewlett, 1986
Правила, причинная модель	Интерпретация	MRS	Kunz, 1983
Прямые цепочки	Диагностика	EXPERT	Lindberg et al., 1980
Правила	Интерпретация	KAS	Alcridge, 1984
»	Диагностика, управление	OPS-7	Masui et al., 1983
»	Диагностика		Expert Systems, 1986, v. 3, N 1
»	Управление		Drasovich, 1983
Фреймы, правила	Интерпретация	RANZ-LISP	Bonasso, 1983; 1984
Правила	Диагностика	OPS-5	Stansfield, 1984
»	Управление		Expert Systems, 1986, v. 3, N 1
Объектно-ориентированный подход, фреймы	»	ZETA-LISP	Kim et al., 1984

Название	Область применения	Назначение	Стадия существования
ATTENING	Медицина	Обучение студентов анестезиологии	Исследовательский прототип
AUDITOR	Юриспруденция	Помощь аудитору в оценке платежной способности клиента	То же
AUTOMECH	Автомобилестроение	Диагностика неисправностей двигателей	Демонстрационный прототип
BABY	Медицина	Контроль за новорожденными в спецкамерах	То же
BATTLE	Военные приложения	Рекомендация по размещению орудий на поле боя	То же
BDS	Электроника	Обнаружение неисправного модуля в цепи сигнала	Исследовательский прототип
BLUE BOX	Медицина	Подбор терапии для больных, страдающих депрессией	То же
Business Plan	Финансовое дело	Коммерческое планирование	Коммерческая
C-13	Химия	Определение состава химических соединений по спектру ядерного магнитного резонанса	Демонстрационный прототип
CAA	Медицина	Распознавание электрокардиограммы	Исследовательский прототип
CADHELP	Электроника	Проектирование цифровых логических схем	То же
CALLISTO	Вычислительная техника	Помощь в распоряжении имеющимися ресурсами	Действующий прототип
CARGuide	Информатика	Оптимизация пути автомобиля на городских улицах	Демонстрационный прототип
CASNET/ Glaucoma	Медицина	Лечение глаукомы	Исследовательский прототип
CATS	Промышленность	Анализ неисправностей локомотивов	Коммерческая
CDX	Вычислительная техника	Анализ дампов аварийного завершения систем	Действующий прототип
CENTAUR	Медицина	Помощь в лечении легочных заболеваний	Исследовательский прототип
CLOT	"	Анализ нарушений свертываемости крови	Демонстрационный прототип
CODE	Вычислительная техника	Определение концептуальной структуры БД	Исследовательский прототип

Представление знаний	Тип задачи	Инструментарий	Источники
Правила	Обучение		Miller, 1983
	Интерпретация	AL/X	Dungan, 1983
Правила в виде сети	Контроль	CSRL	Expert Systems, 1986, v. 3, N 2
Правила	Управление	PASCAL	Rodewald, 1984
»	Диагностика	LES	Slagle et al., 1983
»	»	EMYCIN	Laffey et al., 1984a
»	Планирование		Mulsant et al., 1984
»			AI, 1987
»		INTER-LISP	Gray, 1984
Фреймы, семантическая сеть	Интерпретация	PSN	Expert Systems, 1985, v. 2, N 2
То же	Проектирование	FRANZ-LISP	Cullingford et al., 1982
»	Управление		Expert Systems, 1986, v. 3, N 1
Графическое	»		Sugie et al., 1984
Семантические сети	Диагностика	FORT-RAN	Weiss et al., 1978; 1981
	»		Expert Systems, 1986, v. 3, N 1
	»		То же
Фреймы, правила	»	INTER-LISP	Aikins, 1983
Обратные цепочки	»	EMYCIN	Bernet et al., 1984
Обратные цепочки, правила вывода, основанные на моделирующих правилах и эвристиках, фреймы	Проектирование	LISP	CAIA, 1984

Название	Область применения	Назначение	Стадия существования
Commercial Lerding CONAD	Финансовое дело Электроника	Коммерческие займы Проверка поступления команд и конфигурации компьютерных систем	Коммерческая Действующий прототип
Configure Series CONGEN	Структурная химия	Составление конфигурации компьютеров Определение возможных структур неизвестных соединений	То же Исследовательский прототип
Conphyde	Химия	Выбор аналитической программы для оценки проектов с точки зрения их физических свойств	Демонстрационный прототип
Control Railroad Train CRIB	Промышленность Вычислительная техника	Проверка тормозов железнодорожных составов Обнаружение ошибок и неисправностей	Коммерческая Исследовательский прототип
CRITTER	Электроника	Анализ правильности, своевременности и быстроты срабатывания схем	Демонстрационный прототип
CSS	•	Помощь при планировании перемещений (перестановок) компонентов компьютеров	Действующий прототип
DAA	•	Размещение модулей по алгоритмическим описаниям проекта	Исследовательский прототип
DART	Вычислительная техника	Обнаружение неисправностей в электронных схемах	То же
DAS-LOGIC	Военные приложения Вычислительная техника	Обработка разведанных	»
DELTA	Промышленность	Помощь проектировщикам вычислительных сетей Обнаружение и устранение неисправностей дизельных электровозов	Действующий прототип То же
DENDRAL	Химия	Определение структуры химических соединений по данным о спектре и магнетизме ядра	Промышленная
Detect cracks in billets and direct grinding DFT	Промышленность	Обнаружение трещин в металлических заготовках и шлифовках	Коммерческая

Представление знаний	Тип задачи	Инструментарий	Источник
Алгоритмы теории графов	Диагностика	LISP	AI, 1987
	Управление		Expert Systems, 1986, v. 3, N 1
	Планирование		To же
	Интерпретация	INTER-LISP	Buchanan et al., 1978
	Управление	KAS	Banares-Alcantara et al., 1983
Правила	Диагностика		Expert Systems, 1986, v. 3, N 1
	»		Addis, 1980
	Проектирование	INTER-LISP	Kelly et al., 1982; Kelly, 1984
	Планирование		Expert Systems, 1986, v. 3, N 1
	Проектирование	OPS-5	Kowalski et al., 1983
Фреймы, формулы, исчисление предикатов	Диагностика	MRS	Bennett et al., 1981
	Интерпретация	PASCAL, C	Barth et al., 1983
	Проектирование		Expert Systems, 1986, v. 3, N 1
	Диагностика	LISP	Bonissone et al., 1983
	Интерпретация	INTER-LISP	Gray, 1984
Прямые и обратные цепочки, правила	Диагностика		Expert Systems, 1986, v. 3, N 1
	Интерпретация		

Название	Область применения	Назначение	Стадия существования
DIAG-8100	Электроника	Устранение обнаруженных нарушений правил разработки проекта	Демонстрационный прототип
Diagnose faults on printed circuit boards	"	Обнаружение неисправностей в оборудовании	
DIAGNOSER	"	Обнаружение неисправностей в печатных платах	Действующий прототип
DIGITALIS ADVISOR	Медицина	Выявление наследственных заболеваний сердца	Исследовательский прототип
DIPMETOR ADVISOR	"	Подбор терапии при заболеваниях сердца	Демонстрационный прототип
	Геология	Определение глубинной геологической структуры путем анализа разреза пластов	Исследовательский прототип
DISPATCHER	Промышленность	Составление графика производства и доставки	Коммерческая
DRILLING ADVISOR	Геология	Определение причин поломки бурильных установок в скважинах	Исследовательский прототип
DRUG INTERACTION CRITIC	Медицина	Принятие решений о возможности одновременного назначения нескольких медикаментов	Демонстрационный прототип
ECESIS	Космонавтика	Управление системой жизнеобеспечения экипажа на борту космического корабля	Исследовательский прототип
ECUS	Медицина	Интерпретация ЭКГ	
EDAAS	Информатика	Принятие решения о возможности публикации служебной информации о токсичных веществах	Промышленная
EEG ANALYSIS SYSTEM	Медицина	Анализ электроэнцефалограмм	Исследовательский прототип
EL	Электроника	Анализ схем	То же
ELAS	Геология	Анализ разрезов буровых скважин	"
EMERGE	Медицина	Выработка решений о госпитализации больных в травмопункт	Действующий прототип
ENGINE COOLING ADVISOR	Промышленность	Выявление причин шума в системе охлаждения двигателя автомобиля	Коммерческая

Представление знаний	Тип задачи	Инструментарий	Источник
Фреймы	Проектирование	PROLOG	Horstman, 1983
	Диагностика		Expert Systems, 1986, v. 3, N 1
	»		To же
	»	LISP	Johnson et al., 1981
	»	CWL-1	Swartout, 1977
	Правила, прямые цепочки	INTER-LISP	Davis et al., 1981 Gershman, 1982
	Управление		Expert Systems, 1986, v. 3, N 1
	Правила, обратные цепочки	KS300, S. 1	Hollander et al., 1983
	Фреймы	PROLOG	Roach et al., 1984
	Правила	LISP	CAIA, 1984
Иерархические сети и/или графы, фреймы	Правила		To же
	Диагностика	FORT-RAN	Feinstein et al., 1984; 1985
	»	C	Baas et al., 1984
	»	ARS	Duda et al., 1984 Stallman et al., 1979
	»	EXPERT	Apte et al., 1984a; Apte, 1982
Фреймы	Планирование	PA SCAL	Hudson, 1981;
	Диагностика		Hudson et al., 1984

Название	Область применения	Назначение	Стадия существования
EPES	Авиация	Обеспечение безопасности полетов, обработка информации	Исследовательский прототип
ESCORT	Нефтедобыча	Облегчает пользователю работу с информационными системами, формируя массивы диагностических данных в реальном масштабе времени	То же
EURISKO	Электроника	Проектирование новых видов микроэлектронных устройств	Коммерческая
Experimental Design		Определение требований к проектам и выбор наиболее приемлемого	
EXPER-TAX	Финансовое дело	Учет, ревизия и планирование денежных поступлений, налогов и т. п.	
Expert Navigation	Военные приложения	Навигация для самолетов	Коммерческий прототип
EXPO	Геология	Разведка полезных ископаемых	
FAITH	Космонавтика	Передача телеметрической информации с борта космического корабля	Коммерческая
FALCON	Химия	Определение вероятных причин нарушений процессов на химических установках	Демонстрационный прототип
FAULT-FINDER	Вычислительная техника	Обнаружение неисправностей дисководов	Исследовательский прототип
FG 502—TASP	Электроника	Обнаружение неисправностей	
FOLIO	Экономика	Управление инвестициями	Демонстрационный прототип
FOSSIL	Палеонтология	Идентификация находки	Промышленная
FOREST	Электроника	Обнаружение сбоя программы тестирования	
FUZZY Control System	Промышленность	Регулирование условий работы печи; система подключена к контрольно-измерительным системам печей для отжига цемента	
GA-1	Химия	Анализ структур ДНК	Исследовательский прототип

Представление знаний	Тип задачи	Инструментарий	Источник
Сеть, правила	Управление	ZETA-LISP	CAIA, 1984
Гибкая БЗ		PASCAL	Expert Systems, 1986, v. 3, N 1
	Проектирование	INTER-LISP, TURBO-PASCAL	Lenat, 1982; 1983 Expert Systems, 1986, v. 3, N 2
Фреймы	Планирование, контроль	LISP	Expert Systems, 1986, v. 3, N 3
Правила	Контроль	LISP	Pisano et al., 1984 AI, 1987
Производственные правила, прямые и обратные цепочки, фреймы	Контроль, диагностика		Friedman, 1983; 1984
Правила, прямые цепочки, сеть	Диагностика	LISP	Chester et al., 1984 Expert Systems, 1986, v. 3, N 1
Правила	»	SMALL-TALK-80	Alexander et al., 1984a
Прямые цепочки, правила, линейное программирование	Управление	MRS	Cohen et al., 1983
Деревья, текстовая иерархия		PROLOG	Expert Systems, 1986, v. 3, N 2
Правила	Диагностика	PROLOG	CAIA, 1984
Правила, размытая логика	Управление, контроль		Simmonds, 1985
	Интерпретация	INTER-LISP	Stefik, 1978

Название	Область применения	Назначение	Стадия существования
CALEN	Медицина	Выявление врожденных заболеваний сердца у детей	Исследовательский прототип
GCA	Управление	Составление расписания занятий студентов	То же
GUIDON	Медицина	Обучение студентов противомикробной терапии стационарных больных	»
HANNIBAL	Военные приложения	Оценка ситуации по разведанным о средствах связи	»
HAZARD	Химия	Проверка на канцерогенность вещества	
Headmed	Медицина	Выявление психических заболеваний и выбор медикаментозного лечения	»
Heart Image Interpreter	.	Анализ сердечной деятельности по динамическим характеристикам сердца	Демонстрационный прототип
Headex	Химия	Разработка сетей, требующих минимум энергии при теплообмене	
HI CLASS	Электроника	Определение последовательности операций при сборке ПЭВМ	То же
HT-ATTENDING	Медицина	Помощь в ведении гипертонических больных, информация о новых медикаментах	Исследовательский прототип
HYDRO	Геология	Моделирование физических процессов на водоразделах	То же
Intelligent front-end for complex software	Программирование	«Интеллектуальное» периферийное устройство для сложного ПО	»
Internist I/Caduceus	Медицина	Выявление внутренних заболеваний по истории болезни, симптоматике	Действующий прототип
I&W	Военные приложения	Предсказание возможных конфликтов	Демонстрационный прототип
ICLX	Промышленность	Обнаружение неполадок в процессе фрезеровки стержней	Коммерческая

Представление знаний	Тип задачи	Инструментарий	Источник
Правила, фреймы	Диагностика		Thomson, 1983
Правила с факторами уверенности	Планирование	PROLOG	Valtorta et al., 1984
Правила	Диагностика	MYCIN	Clansey, 1979a; 1983a
	Интерпретация	AGE	Brown et al., 1982
Деревья	Контроль		Expert Systems, 1984, v. 1, N 2
	Диагностика	EMYCIN	Heiser et al., 1978
Семантические сети, фреймы, правила с фактором уверенности	Диагностика		Niemann et al., 1984
Продукционные правила	Конструирование	OPS-4	Expert Systems, 1985, v. 2, N 1 Expert Systems, 1986, v. 3, N 1
Фреймы	Управление (лечение)	LISP	Miller et al., 1984
Правила, семантические сети	Прогнозирование	INTFR. LISP	Reboh et al., 1982 Expert Systems, 1986, v. 3, N 1
Сеть	Диагностика	LISP	Miller et al., 1984
Прямые цепочки, правила	Прогнозирование	INTER-LISP	Kizemidjian et al., 1983
	Диагностика		Expert Systems, 1986, v. 3, N 1

Название	Область применения	Назначение	Стадия существования
IDT	Вычислительная техника	Диагностические тесты для PDP-11/03	Исследовательский прототип
IMACS	Промышленность	Помощь в управлении производством вычислительной техники	Действующий прототип
ISA	Вычислительная техника	Составление графика работы пользователей	Коммерческая
ISIS	Промышленность	Управление производством, составление технологических схем	Исследовательский прототип
IN-ATE	Электроника	Техническое обслуживание осциллографа	То же
INFORM-ART ADVISOR	.	Консультации покупателей компьютеров	
ISS-3	ЭВМ	Анализ и ввод данных в планировщик производства	Коммерческая
JUDITH	Юриспруденция	Решение дел по гражданскому праву	
KARNAK	Электроника	Анализ трудностей процесса волнового паяния при сборке печатных схем	Исследовательский прототип
KARDIO-E	Медицина	Диагноз аритмии сердца по ЭКГ, обучение чтению ЭКГ студентов-медиков	
KNEECAP	Космонавтика	Планирование работы экипажа на борту космического корабля	Коммерческая
LDS	Юриспруденция	Решение дел о платежах	»
Legal Analysys System	То же	Решение дел о преднамеренных гражданских нарушениях и оскорблении действием	Демонстрационный прототип
LES	Космонавтика	Контроль загрузки жидкого кислорода	Исследовательский прототип
LES	Электроника	Техобслуживание электронных приборов	То же
LITHO	Геология	Интерпретация данных о нефтяных буровых скважинах	Исследовательский прототип
MACSYMA	Математика	Действия с алгебраическими выражениями, решение уравнений и т. п.	Коммерческая

Представление знаний	Тип задачи	Инструментарий	Источник
Правила, прямые цепочки	Диагностика	FRANZ-LISP	Schubin et al., 1982
	Управление	OPS-5	O'Connor, 1984
То же	Планирование		O'Connor, 1984; Orciuch et al., 1984
Фреймы и правила	Управление	SRL	Fox et al., 1982; Fox et al., 1983
	Диагностика	FRANZ-LISP	Cantone et al., 1983
	Системное программирование		Expert Systems, 1986, v. 3, N 1
			Datamation, 1986, v. 32, N 5
	Интерпретация	FORTRAN	Popp et al., 1975
	Диагностика	OPS-5	Expert Systems, 1986, v. 3, N 3
Правила	Диагностика, обучение	PROLOG	Expert Systems, 1985, v. 2, N 1
Фреймы	Планирование	INTFR-LISP	Scarl, 1984
Правила	Интерпретация	POSIE	Waterman et al., 1980; Waterman et al., 1981
Семантические сети	»	PSL	Meldman, 1975
Фреймы	Контроль	ZETA-LISP	Scarl, 1984; Scarl et al., 1984
Продукции, правила, фреймы	Диагностика		Expert Systems, 1986, v. 3, N 2
Правила, обратные цепочки	Интерпретация		Bonnet et al., 1983
		LISP	Martin et al., 1971

Название	Область применения	Назначение	Стадия существования
Maintex	Техника	Анализ неисправностей цифровых контролирующих приборов	Коммерческая
MECS-AI	Медицина	Лечение сердечно-сосудистых заболеваний и щитовидной железы	»
MED-1	•	Анализ заболеваний, связанных с болями в области груди	»
MEDICO	•	Ведение больных в офтальмологии	»
MES	Военные приложения	Выявление неисправностей самолета	Демонстрационный прототип
Message Tracse Analyzer	Электроника	Проверка каналов связи в коммутаторах из сотен процессоров	То же
META-NDRAL	Химия	Идентификация по масс-спектральным данным молекулярных структур	Исследовательский прототип
M-1	Медицина	Выявление инфаркта миокарда	Демонстрационный прототип
MIXEP	Вычислительная техника (программирование)	Написание микропрограмм	То же
MODIS (МОДИС)	Медицина	Анализ различных форм гипертонии	»
MOLGEN	Молекулярная генетика	Разработка экспериментов по вегетативному размножению	»
Motor Brush, Designer	Промышленность	Проектирование щеток и пружин для небольших электродвигателей	Коммерческая
Mover	Гражданское строительство	Диагностика неисправностей и рекомендации по их устранению при проектировании систем в аэропортах	
MUD	Геология	Установление свойств бурового газа, определение причин осложнений	Действующий прототип
MUD-MAN	•	Анализ неисправностей при бурении нефтескважин	То же

Представление знаний	Тип задачи	Инструментарий	Источник
	Диагностика	PROLOG	Expert Systems User, 1987, v. 2, N 11
Обратные цепочки, правила	»	LISP	Kaihara et al., 1978; 1982
То же	»	EMYCIN	Puppe et al., 1983
Правила	»	C	Walser et al., 1976; 1977
Прямые цепочки, правила	»	LISP	Ferguson, 1983
	Контроль	PROLOG	Waterman, 1986
	Интерпретация	INTER-LISP	Buchanan et al., 1978
Прямые цепочки, правила	Диагностика	EXPERT	Kastner et al., 1983
Правила-данные	Программирование	PROLOG	Shimizu et al., 1983
Семантические сети, фреймы, правила	Диагностика	LISP	Bokhua et al., 1982
Объектно-ориентированное, фреймы	Планирование	LISP	Stefik, 1981a
	Проектирование		Expert Systems, 1986, v. 3, N 1
	Диагностика	KAS	Expert Systems, 1986, v. 2, N 1
Прямые цепочки, правила	»	OPS-5	Kahn, 1984; Kahn et al., 1984
То же	»		Expert Systems, 1986, v. 3, N 1

Название	Область применения	Назначение	Стадия существования
MYCIN	Медицина	Выбор антимикробной терапии в условиях стационара	Действующий прототип
NAVEX	Космонавтика	Контроль по данным, получаемым от радара, скорости и расположения космического корабля	Исследовательский прототип
NDS	Электроника	Обнаружение неисправностей в общегосударственной коммуникационной сети	Демонстрационный прототип
NEOMYCIN	Медицина	Лечение менингита и других заболеваний	Исследовательский прототип
NEUREX	"	Выявление заболеваний нервной системы	Демонстрационный прототип
NEUROLO— GIST-1	"	Выявление неврологических заболеваний	То же
Nexperts	Издательская деятельность	Верстка газетной полосы	Коммерческая
NPPC	Ядерная энергетика	Определение причины аварий и отклонения от нормального режима работы ядерных установок	Исследовательский прототип
NTC	Вычислительная техника	Решение проблем, связанных со сбоями в компьютерных сетях	Ограниченно действующий прототип
OCEAN SURVEIL LANCE	Военные приложения	Помощь экипажу в определении удаленности судна	Исследовательский прототип
OCSS	Химия	Синтез сложных органических молекул	То же
OCULAR HERPES MODEL	Медицина	Лечение больных лишаем	Демонстрационный прототип
ONCOCIN	"	Ведение онкологических больных в условиях химиотерапии	Исследовательский прототип
PEACE	Электроника	Проектирование электронных схем	Демонстрационный прототип
PEC	Медицина	Ведение больных с угрозой слепоты	Исследовательский прототип
Performance Mentor	Вычислительная техника	Техобслуживание фотолитографического производства схем	

Представление знаний	Тип задачи	Инструментарий	Источник
Правила, обратные цепочки, дерево кон-текстов	Диагностика	LISP	Shortliff, 1976
Правила, фреймы	Контроль	ART	March, 1984a; 1984b
Правила	Диагностика		Williams, 1982; Williams et al., 1983
Правила, прямые цепочки, метаправила	»	INTFR-LISP	Clancey et al., 1981a; 1983b
Правила, прямые и обратные цепочки	»	LISP	Reggia, 1978a
Аналого-геометрическая модель центральной нервной системы	»	LISP	Xiang et al., 1984
6 БЗ	Планирование	ART	Expert Systems, 1986, v. 3, N 1
Правила (алгоритм «здорового смысла»)	Диагностика		Underwood, 1982
	»		Exper Systems, 1986, v. 3, N 1
Правила, прямые цепочки	Контроль	OPS-5	Groundwater, 1984
	Планирование		Corey et al., 1969
Прямые цепочки, правила	Диагностика	EXPERT	Kastner et al., 1982
Обратные и прямые цепочки	»		Shortlife et al., 1981; 1984
Логико-ориентированное	Проектирование	PROLOG	Dincbas, 1980
Правила, прямые цепочки	Диагностика	EXPERT	Kastner et al., 1984
			AI, 1987

Название	Область применения	Назначение	Стадия существования
PIP	Медицина	Ведение истории болезни, диагностика	Исследовательский прототип
PLANT/ds	Сельское хозяйство	Диагностика заболеваний соевых культур	То же
PRIDE	Промышленность	Создание и анализ новых проектов машин	Коммерческая
PROJCON	Информатика	Диагностика и отладка ПО	Демонстрационный прототип
PROSPECTOR	Геология	Поиски залежей полезных ископаемых	Промышленная
PTRANS	Промышленность	Управление производством вычислительной техники	Исследовательский прототип
PUFF	Медицина	Выявление заболеваний легких	Промышленная
Read	Космонавтика	Определение функциональных требований к ПО управления спутниками	Исследовательский прототип
Redesign	Электроника	Проектирование электронных схем	То же
Refine	Программирование	Разработка ПО	Коммерческая
RESEDA	Информатика	Поиск биографических сведений периода средневековья Франции	Демонстрационный прототип
R1	Вычислительная техника	Выбор оптимальной конфигурации ОС VAX	Коммерческая
R1-SOAR	То же	Разработка ОС	Демонстрационный прототип
RADEX	Медицина	Выявление заболеваний печени	Исследовательский прототип
RBMS	Космонавтика	Составление графика работы космического центра	Демонстрационный прототип
RTS	Военные приложения	Классификация кораблей на основании данных радара	Исследовательский прототип
RUBRIC	То же	Доступ пользователя к текстовальной БД	То же
RX	Медицина	Исследование обширных клинических БД	Демонстрационный прототип

Представление знаний	Тип задачи	Инструментарий	Источник
Фреймы	Диагностика	CONNIVER	Kulikowski, 1980
Диагностические правила, индуктивный вывод	»	ADVISE	Michalski et al., 1980a; 1982
	Конструирование		Expert Systems, 1986, v. 3, N 1
Правила	Диагностика	EMYCIN	Waterman, 1986
Правила, семантические сети	Прогнозирование	INTERLISP	Gaschnig, 1981; Gaschnig, et al., 1982
Правила	Управление	OPS-5	Haley et al., 1983
Обратные цепочки, правила	Диагностика	EMYCIN	Aikins et al., 1983
Правила, фреймы с фактором уверенности	Проектирование	KES	Expert Systems, 1986, v. 3, N 2
	»		Mitchell et al., 1983
			AI, 1987
Правила, фреймы	Интерпретация	VSAPL	Zarri, 1984a; 1983
	Проектирование	OPS	McDermott, 1980; 1983
	Конструирование		Rosenbloom et al., 1984
Фреймы	Диагностика	LISP	Waterman, 1986
Правила	Планирование		Marsh, 1984a
»	Интерпретация	PRANZ-LISP	McCune et al., 1983a
»		FRANZ-LISP	McCune et al., 1983b
	»	INTER-LISP	Blum, 1982

Название	Область применения	Назначение	Стадия существования
SACON	Машиностроение	Структурный анализ различных объектов, моделирование их механических свойств	Исследовательский прототип
SADD	Электроника	Проектирование схем цифровых устройств	Демонстрационный прототип
SCENARIO AGENT	Военные приложения	Моделирование поведения в ситуации вооруженного конфликта для «несверхдержав»	То же
SECOFOR	Геология	Консультация по проблемам бурения нефтедобывающих скважин	»
SEQ	Молекулярная биология	Анализ последовательностей нуклеотидов	Коммерческая
Serum protein analysis	Медицина	Анализ белковой сыворотки	»
SIAP	Военные приложения	Идентификация типа океанских кораблей по акустическим данным	Исследовательский прототип
SOPHIE	Электроника	Обучение студентов обнаружению неисправностей в электронных схемах	То же
SPAM	Военные приложения	Анализ фотоснимков аэродромов, определение местоположения объектов	»
PERIL-1	Машиностроение	Оценка возможного ущерба от землетрясения	Демонстрационный прототип
SPERIL-2	То же	То же	То же
SPEX	Молекулярная биология	Разработка сложных лабораторных экспериментов	Исследовательский прототип
SSP	Юриспруденция	Определение, имеет ли право истец на выплату страховой суммы в случае болезни	Коммерческая
STEAMER	Машиностроение	Обучение управлению паровыми силовыми установками	Действующий прототип
STOWAGE PLANNER	Промышленность	Планирование хранения грузов на товарных складах	Коммерческая
SWIRL	Военные приложения	Интерактивное моделирование воздушного боя	Исследовательский прототип

Представление знаний	Тип задачи	Инструментарий	Источник
Обратные цепочки, правила	Моделирование	EMYCIN	Bennet et al., 1978; 1984
Фреймы	Проектирование	LISP	Crinberg, 1980
Правила, прямые цепочки	Моделирование	ROSIE	Schwabe et al., 1982
	Диагностика		Expert Systems, 1986, v. 3, N 1
	»	LISP	Clayton et al., 1981
	Интерпретация		Waterman, 1986
Правила	»	INTERLISP	Drazowich, 1983
	Диагностика	FORT-RAN, INTERLISP	Brown et al., 1975; 1982
Правила	Интерпретация	OPS-5	McKeown, 1984
Правила, прямые цепочки	Диагностика	C	Ishizuka et al., 1981
То же	»	PROLOG	Ogawa et al., 1984a
	Планирование		Iwasaki et al., 1982a
Правила	Интерпретация	X1	Expert Systems, 1986, v. 3, N 1
Объектно-ориентированное	Диагностика	ZETA-LISP	Hollan et al., 1984
	Планирование		Expert Systems, 1986, v. 3, N 1
То же	Моделирование	ROSS	Klahr et al., 1982

Название	Область применения	Назначение	Стадия существования
SYN	Электроника	Разработка электронных схем	Исследовательский прототип
Synergist	"	Помощь в настройке и отлаживании схем, планирование тестирования	Коммерческая
SYSTEM-D	Медицина	Выявление причин головокружения	Исследовательский прототип
TALIB		Разработка ИС	То же
TATR	Военные приложения	Определение целей для атаки в воздухе, выбор наиболее эффективного оружия	»
TAXADVISOR	Юриспруденция	Решение вопроса о налоге и планировании имущества клиентов с большим состоянием	»
TAXMAN	То же	Решение вопроса о налогах	»
Technical risk control	Техника	Решение задач, связанных с техническим риском, в области механического производства, подъемных, гидравлических и механизмов	Действующий прототип
THYROID MODEL	Медицина	Выявление заболеваний щитовидной железы	Демонстрационный прототип
TOMSTONE	Химия	Анализ данных масс-спектрометра	Исследовательский прототип
Transisto Sizing System	Электроника	Разработка ИС	То же
Tropicalaid	Медицина	Помощь при диагностике в амбулаториях в тропических условиях, выбор лечения	Действующий прототип
Troubleshoot Problems in Soup Cookers	Промышленность	Техническое обслуживание печей, предсказание неисправностей	Коммерческая
TVX	Вычислительная техника	Обучение пользованию ОС	»
TWIRL	Военные приложения	Интерактивное моделирование наземного боя	Исследовательский прототип
UCW Expert	То же	Обучение техобслуживанию самолетов	То же

Представление знаний	Тип задачи	Инструментарий	Источник
Правила	Проектирование		Hayes et al., 1983
—	Диагностика	Flavors	Expert Systems User, 1987, v. 2, N 11
Фреймы	»	KM	Reggia et al., 1983
Прямые цепочки	Проектирование Управление	OPS-5 POSIE	Waterman, 1986 Callero et al., 1984
Правила	Планирование	EMYCIN	Michaelson, 1982; Michaelson et al., 1983
Фреймы	Интерпретация	AIMDS	Kedar-Cabelli, 1984
Правила	Диагностика	PASCAL	Expert Systems, 1986, v. 3, N 3
Прямые цепочки, правила	»	EXPERT	Kulikowski et al., 1980
	Интерпретация	KEE	Wong et al., 1983; 1984
Объектно-ориентированное, фреймы	Проектирование	LOOPS	Foyster, 1984
Фреймы (400 заболеваний)	Диагностика	PASCAL	Expert Systems, 1986, v. 3, N 1
	»		To же
	Обучение		»
Объектно-ориентированное	Моделирование	ROSS	Klahr et al., 1984
	Диалектика	Xi Plus	Expert Systems User, 1987, v. 2, N 11

Название	Область применения	Назначение	Стадия существования
VM	Медицина	Лечение послеоперационных больных	Исследовательский прототип
WHEAT CO-UNSELLOR	Сельское хозяйство	Выявление заболеваний озимой пшеницы	Ограниченно действующий прототип
WHEEZA	Медицина	Лечение легочных заболеваний	Исследовательский прототип
WILLARD	Метеорология	Предсказание гроз в центральной части США	Демонстрационный прототип
XCON	Вычислительная техника	Создание конфигурации вычислительной системы, выбор компонентов	Промышленная
XSEL	То же	Консультация при покупке компонентов ОС	»
YES	•	Измерение характеристик и планирование производительности ЭВМ	Исследовательский прототип
YES/MVS	•	Контроль и управление операционной системой	Коммерческая
ДАМП	Программирование	Анализ дампов телемонитора	Промышленная
КОНСУЛЬТАНТ-2	Медицина	Острые заболевания брюшной полости	Исследовательская
ЛЕДИ-Z	•	Выявление заболеваний	»
МОДИС	•	Анализ причин гипертонии	»
ОБЕЗБОЛИВАНИЕ	•	Определение метода анестезии при стоматологических вмешательствах	Исследовательский прототип
ПЛОТИНА	Гидростроительство	Анализ состояния гидросооружений	Исследовательская
СОНЭТ	Электроника	Локализация неисправностей в электронных схемах	»
ЭСПЛАН	Нефтепереработка	Распределение сырья по установкам первичной переработки нефти	»

Представление знаний	Тип задачи	Инструментарий	Источник
Правила	Контроль	INTER— LISP, EMYCIN	Fagan et al., 1979
	»		To же
Обратные и прямые цепочки	Диагностика	RLL	Smith et al., 1980; 1984
Правила	Прогнозирование	RULE- MASTER	McKeon et al., 1984
»	Конструирование	OPS-5	McDermott, 1980; 1981; 1984
»		OPS-5	McDermott, 1982
			Datamation, 1986, с. 32, N 5
Правила (прямые цепочки)	Контроль, управление	OPS-5	Grieasmer, et al., 1984; Schor, 1984
Правила, подпрограммы	Диагностика	ЭКСПЕРТИЗА	§ 18.2
Правила, глубинные и поверхностные знания	Диагностика	МЕ- ДИФР-3, Реляп	Самсонов, 1987
Фреймы	»	ФРЛ	§ 18.4
Фреймы, правила	»	СПЭЙС	§ 17.1
Правила	»	ЭКСПЕРТ	§ 17.2
»	»	ЭКСПЕРТИЗА	§ 18.1
»	»		§ 18.3
»	Планирование	Пролог, SM-1600, IBM PC	Гулько, 1987

лить организмы, являющиеся причиной заболевания, и назначить антибиотики избирательного действия.

Система обладает сведениями о возможной структуре проблемной области, конкретизируемой в ходе консультации. В системе знания о предметной области хранятся в виде троек атрибут-объект-значение, снабженных коэффициентом определенности, например (идентичность организма KLEBSIELLA) 0.25. Тройки группируются по объектам в соответствии с деревом контекстов, задающим возможную иерархию объектов в предметной области. Так, в дереве контекстов определено, что у пациента может быть взята одна или несколько культур, каждая из которых может содержать один или несколько организмов. Дерево контекстов используется для управления ходом консультации, создания в рабочей памяти новых объектов, отслеживания иерархических зависимостей между объектами (тройками) и использования их в ходе сопоставления. Отметим, что каждой тройке системы MYCIN соответствует утверждение в смысле системы PROSPECTOR.

Процесс решения задачи состоит не только в вычислении определенностей утверждений, как в PROSPECTOR, но и в определении самих утверждений (троек) и оценки их правдоподобности. Определение значений атрибутов и вычисление их коэффициентов определенности осуществляется на основе правил, описывающих процесс решения задачи. Для манипулирования определенностями используются эмпирические формулы, которые могут быть проинтерпретированы в терминах теории вероятностей и теории размытых множеств. Существует их связь с формулами, используемыми в системе PROSPECTOR. В ходе решения задач система на основе дерева контекстов строит представление о структуре предметной области, определяя номенклатуру входящих в нее объектов и их иерархическое подчинение. Основная управляющая стратегия системы — планирование вывода от целей в глубину с вычислением достоверных значений от данных. В ходе решения задачи используется исчерпывающий поиск.

Система способна давать объяснения, ПОЧЕМУ потребовалась запрашиваемая информация и КАК получен результат. Объяснения строятся на основе хранящейся истории работы системы и интерпретации правил. Система содержит около 800 правил. Создание системы MYCIN привело к развитию работы в следующих направлениях: автоматизации процесса формирования БЗ и повышения роли эксперта в создании системы (TEIRESIAS); модифицирования системы для ее использования в обучении студентов методам диагностики и терапии (GUIDON, NEOMYCIN); обеспечения обработки временной информации и совершенствования структуры управления (ONCOCIN).

Системы PROSPECTOR и MYCIN решают задачу классификации в условиях недостаточности используемых данных и знаний. В обеих системах выбор решения осуществляется из относительно небольшого обозримого пространства решения, и проблема стоит не в невозможности его перебора, а в отсутствии достоверных сведений о том, как сделать выбор.

Система R1 предназначена для определения конфигурации компонентов ЭВМ VAX 11/780, удовлетворяющей требованиям заказчика. В состав ЭВМ VAX 11/780 входит 420 компонентов, каждый из которых может иметь до 8—10 характеристик. Естественно, что число возможных конфигураций, построенных из этих компонентов, достаточно велико. Система выполняет следующие функции: определяет, не содержит ли заказ пользователя несовместимых компонентов, и выявляет недостающие; выдает конфигурацию ЭВМ VAX 11/780, которая используется при установке системы заказчику; учитывает обусловленные заказчиком ограничения. До создания системы решением этих задач на фирме занимались квалифицированные специалисты, тем не менее часто допускавшие ошибки. Система достигла коммерческой стадии существования, объем ее БЗ 3000 правил языка OPS-5. Система работает в интерактивном режиме. Средства объяснения реализованы в виде заранее записанных текстов. Реализация системы на языке OPS-4, а затем OPS-5 определила управляемую данными стратегию управления. Эффективность решения задач вызвана тем, что

каждый шаг решения выполняется только тогда, когда для этого имеется достаточная информация, чтобы его можно было сделать точно и в дальнейшем никогда не пересматривать. Поэтому процесс проектирования сводится к постоянному расширению частичной конфигурации. Указанный прием имеет отношение как к применению каждого правила, так и к решению каждой из шести подзадач, на которые разбита исходная задача. Примечательным в системе является еще и то, что для управления порядком решения задач не требуется ввода дополнительных формализмов, кроме правил, имеющихся в языке OPS-5. Управление решением задачи осуществляется чисто продукционным способом. Переход к новой задаче отмечается устранением из рабочей памяти контекста старой задачи и помещением контекста новой задачи. Поскольку все правила в условии содержат контекст задачи, к которой они относятся, то смена контекстов в рабочей памяти осуществляется обычными для языка OPS-5 правилами. Усовершенствованная со временем система R1 получила название XCON. Система R1 (XCON) является примером того, как сложная, по сути, задача проектирования (преобразования) при наличии соответствующих знаний может быть решена простыми эффективно реализуемыми методами.

14.3.2. ЭС, решающие задачи доопределения. Одной из немногих систем, решающих задачу доопределения, является система EL. Система EL предназначена для выяснения неизвестных параметров электрической цепи по ее структуре и по значениям известных параметров (напряжение и ток). Расчеты осуществляются по формулам для линейных элементов электрической цепи. При наличии в цепи нелинейных элементов (диоды, транзисторы) они с помощью линейно-кусочной аппроксимации сводятся к линейным элементам. Однако разные участки нелинейной характеристики необходимо аппроксимировать различными линейными уравнениями. Выбор уравнения зависит от параметров цепи в точке, где находится нелинейный элемент, а эти параметры и требуется определить, и, следовательно, аппроксимирующее уравнение, необходимое для анализа, неизвестно. Система доопределяет состояние предметной области, выдвигая предположение о том, на каком участке характеристики находится нелинейный элемент, и проводит анализ исходя из этого предположения. Если предположение входит в противоречие с другими параметрами, то оно пересматривается и удаляются все его следствия. Если сделано много предположений, возникают проблемы с определением того, какие предположения являются причиной данного противоречия, т. е. какие предположения надо пересматривать. Кроме того, надо следить за тем, чтобы не выставлять заново ранее отвергнутые комбинации предположений. Механизмом выставления и пересмотра предположений является управляемый зависимостями бэктрекинг [Baas et al., 1984; Попов, 1987]. Выставление и пересмотр предположений могут быть полезны не только как средство повышения эффективности решения задач, но и как удобная форма формализации знаний эксперта, отличных от «позитивных» знаний, которые традиционно должен вводить эксперт в случае чисто продукционных систем.

Для решения задач доопределения в ЭС реализуются системы поддержки истинности, обеспечивающие генерацию предположений, их пересмотр и поддержание непротиворечивости получаемых гипотетических альтернативных миров. Наиболее известные схемы поддержания истинности реализованы в виде систем TMS и ATMS [De Kleer, 1986].

14.3.3. ЭС, решающие задачи преобразования. Примером ЭС, решающей задачи преобразования, является система VM. Система предназначена для слежения за состоянием послеоперационных больных, которым необходимо применение аппарата искусственного дыхания. Система определяет тип используемого аппарата, режимы работы в зависимости от развития заболевания и реакции организма на терапевтические воздействия. Система создана на основе методов системы MYCIN и использует частный подход к работе с изменяющимися во времени данными. Представления системы о времени ограничены лишь текущим и предшествующим состояниями. Знания в системе представляются в виде следующих типов правил: перехода; инициализации; состояния; лечения. Система

постоянно получает новые показания приборов и запускает все свои правила. Правила перехода определяют моменты смены состояний больного, когда необходимо изменить порядок слежения за ним. Моменты смены состояний определяются по выходу показателей за заранее установленные границы. С помощью правил инициализации устанавливается новый контекст, т. е. новые ожидаемые значения. После установления контекста правила состояния определяют новое поведение системы до тех пор, пока контекст не изменится. Несмотря на то, что показания приборов меняются все время, система осуществляет изменение контекста только тогда, когда оно будет достаточно обосновано по правилам перехода. Ограниченность подхода, как уже говорилось, состоит в том, что при смене состояний учитывается информация только о двух смежных состояниях пациента.

Работы по созданию систем, имеющих более глубокие представления о времени и моделирующие деятельность активных агентов, в настоящее время находятся в стадии исследований.

Глава 15.

Отечественные инструментальные средства для персональных ЭВМ и мини-ЭВМ

15.1. Исследовательская система СПЭИС

О. В. Ковригин

15.1.1. Назначение и структура. С точки зрения разработчика экспертной системы (ЭС) СПЭИС можно рассматривать как среду проектирования ЭС с открытой архитектурой, в которой ему доступны три уровня.

Первый уровень представляет собой оболочку ЭС. На первом уровне СПЭИС предоставляет разработчику широкие возможности описания и анализа исследуемых объектов, их отношений и правил в интерактивных режимах. Разработчику достаточно ознакомиться с общей идеологией построения ЭС, изучить язык представления знаний и механизм вывода решения, принятые в СПЭИС; ему доступны все вспомогательные функции среды: отладка базы знаний (БЗ), объяснения и др., что облегчает процесс построения и доводки ЭС до действующего прототипа.

Второй уровень представляет собой объектно-ориентированный язык программирования, в котором доступны такие примитивы, как объект (фреймы, концепция, схема), продукционные правила, возможно конструирование различных языков представления знаний. Квалификация разработчика, который обращается ко второму уровню СПЭИС, должна быть выше, чем у пользователя оболочки ЭС. Разработчик должен быть специалистом в области ИИ, иметь опыт создания языков представления знаний, владеть навыками программирования на языке Лисп.

Третий уровень среды — язык Лисп (точнее, его диалект muLISP) и программные функции СПЭИС. На этом уровне разработчику доступно все программное обеспечение СПЭИС, и он может его как модифицировать, так и реализовать новые возможности.

15.1.2. Представление знаний и вывод решений. Представление знаний в СПЭИС базируется на принципах распределенного представления знания.

Вся совокупность знаний эксперта логически разбивается на подмножества, которые группируются вокруг некоторых понятий, называемых в системах ИИ объектами, фреймами, концепциями.

Основу представления в СПЭИС составляют концепции — обособленные структуры знаний (декларативные знания, процедуры и продукции), связанные

между собой различными отношениями. Язык концепций в СПЭИС, сохраняя все возможности представления знаний в виде правил, позволяет структурировать систему продукций, использовать эвристические знания экспертов о наилучших путях поиска решений, записывая эвристики в виде правил.

В качестве механизма вывода решений используется схема, напоминающая «blackboard» [Aiello et al., 1981a]. При такой схеме любое действие ЭС по отысканию решения заключается в выполнении некоторой конкретной работы на каждом шаге (заполнение слота концепции, проверка истинности правила, активизация новой концепции и т. п.). Механизм выполнения любой работы один и тот же, что позволяет сохранять модульный и независимый принцип организации вычислений. А это является основой быстрой модификации, адаптации и расширения СПЭИС.

Ниже будут рассмотрены средства, которые предлагаются пользователю на первом уровне СПЭИС — на уровне оболочки.

Система СПЭИС предназначена в основном для решения задач классификации на основе экспертных знаний. К подобным задачам можно отнести большинство диагностических задач из различных областей человеческой деятельности.

Анализ природы различных задач диагностики позволяет условно выделить три типа знаний, которые необходимы системе для достижения высокого уровня функционирования.

1. Знания эксперта о классификационных (или диагностических) признаках, на анализе которых строится решение задачи. Классификационные признаки представляют собой некоторые понятия проблемной области, которые связаны между собой отношениями и, следовательно, образуют фрагментарные сети — кластеры. К декларативным знаниям о диагностических признаках можно отнести информацию об области определения признака, условиях его существования, ассоциативных связях с другими признаками, характеристиках понятия. Процедурная часть знаний о признаках может относиться к способу определения значения признака или его характеристик и т. п.

2. Знания, которыми пользуется эксперт для представления промежуточных или конечных диагностических состояний. Эти знания позволяют на основе анализа определенных подмножеств классификационных признаков оценить, принадлежит ли исследуемый объект или ситуация некоторому диагностическому состоянию.

3. Знания, которые помогают отыскивать наиболее перспективные пути решения задачи в зависимости от текущего контекста. Под текущим контекстом понимается не только совокупность означенных классификационных признаков и промежуточных заключений, но (это особенно следует подчеркнуть) и предыстория поиска решения, удачные и неудачные ходы в процессе работы. Эти знания являются эвристиками, позволяющими сокращать пространство поиска, обеспечивать естественность (с точки зрения эксперта) размышлений, оставляя в тени ненужные знания и активизируя наиболее подходящие.

Исходя из вышеприведенной классификации знаний, в СПЭИС принята трехуровневая организация БЗ.

На нижнем уровне иерархии располагаются концепции, предназначенные для представления знаний экспертов о классификационных признаках, — так называемые Р-концепции. Они образуют связанные кластеры признаков. С помощью структур первого уровня моделируются основные понятия проблемной области.

Концепции второго уровня — Н-концепции — предназначены для представления знаний о классификационных состояниях. Каждая Н-концепция — это мини-эксперт по определенной проблеме, и, следовательно, в рамках каждой структуры собраны все знания, необходимые для ее решения. Каждая Н-концепция может подключаться к решению общей задачи в зависимости от текущего контекста, формируя более или менее общие заключения. Подобные заключения могут использоваться другими Н-концепциями при решении их задач, а также составлять часть окончательного диагноза ЭС. Каждая Н-концеп-

ция в общем случае ориентирована на сбор и анализ необходимой ей информации с целью выработки некоторых новых утверждений. Такой подход позволяет разработчикам ЭС осуществлять поэтапное построение и отладку БЗ, где на каждом этапе возможно формирование одной Н-концепции независимо от остального наполнения БЗ.

Однако мало сформировать мини-экспертов для решения отдельных подзадач. Необходимо задать некоторую общую схему подключения тех или иных Н-концепций второго уровня к решению задачи. Для этой цели используются эвристические знания экспертов о том, как искать решение наилучшим образом (т. е. какие Н-концепции второго уровня и в каком порядке необходимо привлечь к решению в текущем контексте). Третий уровень представляет собой структуры, содержащие знания о том, когда и какие знания лучше использовать. Эти структуры, также называемые Н-концепциями, содержат информацию о том, что разумно делать в некоторых типичных ситуациях: какие классификационные признаки необходимо определить и какие из структур второго уровня следует привлечь к решению задачи.

Концепции второго и третьего уровней отличаются друг от друга только принципом объединения знаний внутри них. Если на втором уровне знания группируются по принципу отнесения их к одному классификационному состоянию, то на третьем уровне основой объединения является некоторая контекстная ситуация, из которой могут быть различные альтернативы направлений поиска. Н-концепции третьего уровня можно рассматривать как «мета-экспертов».

Рассмотрим непосредственно язык представления знаний в СПЭИС с помощью Р- и Н-концепций. Как Р-, так и Н-концепции представляют собой информационные структуры, в которых информация разнесена по слотам (информационным подструктурам).

В оболочке ЭС СПЭИС Р-концепции могут иметь фиксированный набор слотов. Разрешается оперировать и произвольными слотами для характеристики понятий. Кроме того, благодаря строгой модульности и простоте обработки слотов множество фиксированных слотов легко расширить.

Фиксированный набор слотов Р-концепции может быть полезен практически в любой предметной области.

Каждое понятие в СПЭИС должно иметь уникальное имя. С точки зрения организации БЗ желательно, чтобы это имя не было очень длинным (хотя физические ограничения на длину атома составляют 256 символов). Поэтому структуры, соответствующие классификационным признакам, можно обозначать некоторыми идентификаторами, например P1, P2, ..., Pn или OBJECT001, ..., OBJECT00n. Часто в процессе вывода решения возникает необходимость запрашивать значение классификационного признака у пользователя. Чтобы запрос системы был понятен пользователю, необходимо сформулировать его на естественном языке (ЕЯ). Этой цели служит слот PNAME (далее слоты и процедуры будут выделяться в тексте большими буквами), который содержит ЕЯ-версию названия понятия, описываемого Р-концепцией.

Понятия, описываемые Р-концепциями, могут принимать некоторые значения в контексте решения конкретной задачи.

По значениям классификационных признаков система выводит заключения. Для описания возможных значений признаков служит слот POSSIBLE-VALUE.

Важное значение имеет слот RESTRICTION, хранящий информацию об ограничениях на возможные значения признака. Если признак может принимать значения из некоторого множества атомов, причем одновременно несколько значений, и необходимо отметить, что некоторые значения не могут встречаться одновременно, то в слоте записываются пары несовместимых значений. Если параметр числовой, то в слоте RESTRICTION разрешается задать числовой интервал его значений.

Слоты POSSIBLE-VALUE и RESTRICTION используются непосредственно для полного задания значений отдельного признака. Однако, поскольку часто признаки связаны между собой, необходимо задавать ограничения

на значения одного признака в зависимости от того, какое значение принял другой (или другие). Этой цели служит слот **FRESTRICTION**.

В СПЭИС предусмотрена возможность прямого указания на способ определения значения признака. Для этого используется слот **VALUE**. Если он пуст (или отсутствует), что это означает, что значение признака должно запрашиваться у пользователя ЭС. Если же слот непуст, то он содержит некоторый список работ (процедур и продуктов), определяющий схему вывода значения. Если слот **VALUE** непуст, но предпринятые согласно записанной там информации шаги не привели к установлению значения искомого признака, то оно запрашивается у пользователя.

В СПЭИС предусмотрена обратная операция — установление значения некоторым признакам сразу после определения искомого. Для этого служит слот **SET-VALUE-F**, содержащий перечень работ по установлению значений признаков в зависимости от значения рассматриваемого.

В Р-концепциях предусмотрен еще один слот, имеющий фиксированное использование, — слот **ONLY-FOR**. Его телом является логическое условие, истинность которого означает, что данная Р-концепция может рассматриваться в текущем контексте. При ложности условия признак выключается из дальнейшего рассмотрения. Если этот слот существует, он всегда обрабатывается первым при анализе концепции. С помощью слота **ONLY-FOR** удобно представлять знания о том, что некоторые признаки имеют смысл, только если ряд других принял определенные значения.

Классификационные признаки, используемые экспертом, группируются им в ассоциативные кластеры, т. е. эксперт не выясняет значение одного признака, а обрабатывает сразу целый «куст» понятий. Для представления ассоциативных связей параметров используется слот **LINK**, который содержит имена Р-концепций, связанных с данным.

В СПЭИС предусмотрен ряд вспомогательных слотов. Слот **NORMAL-VALUE** содержит информацию о том, какие значения классификационных признаков являются типичными. Другой вспомогательный слот **COMMENT** служит для хранения справочной информации о признаке и его возможных значениях. Данную информацию пользователь может использовать для пояснения задаваемых вопросов и предлагаемых ответов.

Общая структура слотов с фиксированными правилами обработки для Р-концепции может быть представлена в виде

[DS <идентификатор>	
PNAME	<текст>
ONLY-FOR	<логическое условие>
POSSIBLE-VALUE	<список/«число»/«строка»>
RESTRICTION	<пары элементов/предикаты>
FRESTRICTION	<предикаты>
VALUE	<процедуры/продукции>
SET-VALUE-F	<процедуры/продукции>
LINK	<список имен Р-концепций>
NORMAL-VALUE	<атом/список>
COMMENT	<текст>]

где функция **DS** служит для определения концепций.

В процессе функционирования ЭС различным мини- и метаэкспертам БЗ будет требоваться информация о классификационных признаках. Запрос на определение значения признака в Н-концепциях формируется следующим образом: (GET: P_i P_j), где GET: — имя работы, а P_i, P_j — аргументы, указывающие, какие классификационные признаки необходимо определить. Работа GET: выполняется согласно правилам, которые определены в теле функции GET. Сначала выясняется, известно ли значение искомого признаков. Они могут быть известны, если ранее подобная работа была сформирована какой-либо концепцией. Если значение известно, то работа не выполняется. Если обращение к искомой концепции происходит в первый раз, то производится активизация

Р-концепции соответствующего признака. Это означает, что анализируются соответствующие слоты Р-концепции.

Первым анализируется слот ONLY-FOR. Процесс определения истинности или ложности логического условия этого слота может привести к формированию новых работ GET для установления значений тем признакам, которые входят в условия, но неизвестны к настоящему моменту. В случае ложности условия рассмотрение Р-концепции прекращается и его значение не определяется.

В случае истинности условия существования признака происходит обращение к слоту VALUE. Если он пуст, то значение признака запрашивается у пользователя. Получив ответ пользователя, программа проверяет его соответствие ограничениям, которые находятся в слотах RESTRICTION и FRESTRICITION. Если слот VALUE непуст, то происходит его обработка. В слоте могут находиться процедуры и продукции. Процедуры интерпретируются как требования выполнить определенную работу, а продукции представляют собой выражение вида «если<условие>, то <действие>», где <действие> — это работа, выполнение которой происходит, только если <условие> истинно в текущем контексте.

Для работы со значениями признаков в СПЭИС определен набор процедур:

(SET: <Р-концепция> <значение>) — устанавливает или переустанавливает значение;

(ADD: <Р-концепция> <значение>) — добавляет значение к уже имеющимся значениям Р-концепции;

(SUBST: <Р-концепция> FROM <старое значение> TO <новое значение>) — заменяет уже имеющееся значение на новое.

Список подобных процедур может быть расширен простым доопределением новых функций.

При решении многих задач приходится иметь дело с описанием понятий, которые могут составлять иерархические сети. Каждому понятию свойственны свои характеристики. Для представления таких декларативных знаний о понятиях проблемной области используются фреймы с присоединенными процедурами, обеспечивающие механизм наследования свойств по иерархии понятий. Представление характеристик понятий производится на основе указания определенных атрибутов и их значений для Р-концепции, описывающей признак. Имена атрибутов соответствуют характеристикам, а значения атрибутов могут быть либо константами, либо некоторой процедурой (или списком процедур), с помощью которой должно определяться значение атрибута.

Определив таким образом понятия, можно сформировать в процессе поиска решения конкретные примеры некоторых понятий на основе прототипов. Очевидно, что наследование свойств понятиями не следует ограничивать одним уровнем иерархии понятий. В СПЭИС наследование свойств возможно со всех уровней.

Для формирования конкретного примера понятия в языке предусмотрены две процедуры LET: и CREATE-INSTANCE:.

В СПЭИС разрешается такое определение понятий, когда значением слота может быть также пример некоторого понятия.

Продукции в СПЭИС являются одним из основных элементов представления знаний. С их помощью удобно представлять знания экспертов, выражаемые в виде правил (правила отнесения к классификационным состояниям, условный вывод значений признаков, эвристические правила поиска решений).

Традиционная форма продукции Если <условие> ТО <действие> интерпретируется в СПЭИС следующим образом. Оценка левой части продукции <условие> происходит на основе анализа текущего контекста. Условие продукции строится как логическая комбинация некоторых предикативных функций, связанных операциями AND, OR и NOT.

Правая часть продукции и <действие> трактуется в СПЭИС как список работ, которые необходимо выполнить, если левая часть продукции имеет зна-

чение TRUE. Часть работ уже приведена выше, остальные рассмотрим при описании Н-концепций.

Каждая Н-концепция с точки зрения механизма, осуществляющего вывод, является набором инструкций. Н-концепция содержит процедуры обращения к Р-концепциям, что позволяет организовать сбор необходимой информации (а следовательно, и диалог с пользователем). Включение продуктов в рамки Н-концепции приводит к возможности представления правил принятия решений экспертом в зависимости от собранной информации. Общая структура Н-концепции:

[DS <идентификатор>	
PNAME	<список>
PROTOTYPE	<список>
INHERRITORS	<список>
ONLY-FOR	<логическое условие>
BODY	<процедуры/продукции>]

Слот ONLY-FOR используется для указания контекста, в котором Н-концепция может включаться в рассмотрение (активизироваться). Это не означает, что если логическое условие слота выполняется, то концепция активизируется. Указание на активизацию Н-концепции должно формироваться некоторой другой Н-концепцией. Если поступает команда на такую активизацию, то слот ONLY-FOR проверяет, может ли рассматриваться указанная концепция. Слоты PROTOTYPE и INHERRITORS служат для представления родовидовой информации о Н-концепциях.

Информация о иерархии Н-концепций позволяет реализовать некоторые разумные экспертные стратегии в ходе вывода решения. Знания о иерархии Н-концепций позволяют реализовать стратегии поиска вширь и вглубь в пространстве Н-концепций. Желаемая стратегия определяется разработчиком на этапе проектирования конкретной ЭС.

Ядром Н-концепции является слот BODY, содержащий процедуры и продукты, выполнение их может приводить к активизации Р- или Н-концепции, к установлению заключений о классификационном состоянии, описываемом данной Н-концепцией, и значении контрольных признаков.

Процедуры и продукты слота BODY выполняются последовательно, до тех пор, пока все они не будут проинтерпретированы либо не будет выполнена процедура STOP;; предписывающая останов обработки слота в том месте, где встретилась эта процедура. В последнем случае система прекращает рассмотрение текущей Н-концепции и переходит к дальнейшему выполнению работ из накопившегося списка. Позже происходит возврат к прерванной Н-концепции (она снова становится активной), и процесс оценки слота начнется с той точки, где произошло прерывание.

Обращение Н-концепции к Р-концепциям вызывает активизацию кластеров понятий на первом уровне БЗ, поэтому запрос одного признака из Н-концепции может привести к вычислению некоторого множества признаков.

При анализе слота BODY выполнение работы (ACTIVATE: H1 ... Hk ... Hn) приводит к активизации новых Н-концепций. Однако активизация не происходит немедленно, а откладывается, пока обработка слота не будет прекращена. Следовательно, есть возможность либо накопить список гипотез, подлежащих активизации, и выполнить его после завершения анализа всех процедур или продуктов слота, либо сразу активизировать новую Н-концепцию, записав вслед за процедурой ACTIVATE: процедуру STOP;. Если концепция должна быть активизирована, но после ее обработки не должно происходить возврата в ту концепцию, из которой она была вызвана, необходимо использовать процедуру (ACTIVATE: RETURN H1 ... Hk ... Hn).

Разрешается активизировать Н-концепции и из Р-концепций, например, когда эксперт обладает эвристическим правилом рассмотрения гипотезы на основе анализа значений одного диагностического признака. В этом случае в Р-концепции такого признака вносится слот с фиксированной процедурой обработки

STR-CONTROL. Он может содержать продукции и процедуры, вызывающие активизацию Н-концепций. Вопрос о том, как лучше организовывать активизацию Н-концепций из Р- или Н-концепций, следует решать применительно к конкретной задаче.

Для установления заключений о классификационных состояниях используется процедура (**DIAGNOSE: weight text**). После ее выполнения к текущему контексту добавляется новое заключение, сформулированное в виде текста (аргумент **text**). Аргумент **weight** — целое число, отражающее ранг заключения. Пока Н-концепция подвергается анализу, все заключения, найденные в ней, помещаются во временную память. Заключения с меньшим рангом могут вытесняться из этой памяти текстами с большим рангом. Если в процедуре отсутствует аргумент **weight**, то тексты таких заключений добавляются безусловно во временную память. Это позволяет формировать общее решение по концепции, соединяя разрозненные тексты. После окончания (или прерывания) работы с Н-концепцией заключения из временной памяти поступают в общедоступную.

В ходе анализа Н-концепции может сложиться ситуация, при которой становится очевидным, что рассмотрение концепции необходимо прекратить и не возвращаться к ней ни при каких обстоятельствах. Закрытие концепции и исключение ее из дальнейшего рассмотрения навсегда можно осуществить с помощью выполнения работы (**CLOSE: Hi**).

Язык Н-концепций позволяет описать соответствующую модель процесса принятия решений экспертом в каждой конкретной проблемной области и построить необходимый диалог ЭС с пользователем.

Работа ЭС должна начинаться с анализа Н-концепции, являющейся «мета-экспертом», который организует сбор начальной информации и на основе ее анализа распределяет работу между другими мини- и (или) метаэкспертами. Внешним результатом анализа каждой Н-концепции может быть выработка заключений и активизация новых Н-концепций. Процесс поиска решения будет происходить до тех пор, пока список Н-концепций, подлежащих рассмотрению, не окажется пуст или в одной из Н-концепций не выполнится процедура **GSTOP**. В результате работы ЭС набирается совокупность решений по отдельным Н-концепциям, которая сообщается пользователю.

15.1.3. Средства проектирования и отладки. В состав СПЭИС включены средства, обеспечивающие процесс проектирования и отладки ЭС. Они включают следующие блоки: проверки и отладки БЗ; графической визуализации знаний; ввода-вывода информации; прерывания и трассировки ЭС; анализа состояния системы после сеанса вывода решений.

Первый блок обеспечивает показ и проверку Р- и Н-концепций. Концепции могут быть показаны на экране дисплея в различном виде: 1) в том, в каком они хранятся внутри системы; 2) в виде, когда на место вхождений идентификаторов признаков и Н-концепций подставляются соответствующие им тексты из слотов **PNAME**, при этом текст описания концепций приближается к тексту на естественном (русском) языке, что упрощает процесс понимания описаний; 3) в виде, когда в тексте описания концепции присутствуют необходимые идентификаторы, а в скобках приводится их расшифровка.

Анализ концепций позволяет выявить возможные ошибки в тексте их описания. Проверка проводится с целью выявления ссылок на неопределенные Н- и Р-концепции, на неопределенные процедуры, на неправильное указание аргументов процедур и т. д. Например, предикаты условий продукции проверяются на возможность их принципиального выполнения.

Блок графической визуализации предназначен для наглядного представления связей между концепциями, находящимися в БЗ. По запросу пользователя на экране дисплея может быть цветное изображение сети признаков БЗ с корневой вершиной, указанной пользователем. Направление стрелок на подобной диаграмме позволяет легко проанализировать суть связей. Подобные изображения можно получить и для Н-концепций. В этом случае рисунок изображает каждую Н-концепцию в виде «паука», отмечая связи двух видов, запрос Р-концеп-

ций из Н-концепций и альтернативы активизации других Н-концепций. Такой режим позволяет одним взглядом охватить дерево выводов, проанализировать его и принять соответствующие решения по модификации базы знаний.

Блок ввода-вывода информации позволяет осуществлять общение с файловой системой и сохранять результаты работы со СПЭИС.

Средства СПЭИС разрешают на этапе разработки БЗ производить тестовые решения задачи. Вывод решения можно начать с любой Н-концепции, указав ее как начальную вершину вывода. Модульность и независимость Н-концепций обеспечивает такую возможность. На этапе вывода часто возникает необходимость прервать этот процесс и проанализировать текущее состояние системы. Пользователя может интересовать, каким путем ЭС пришла в текущее состояние, какие признаки получили значения и почему. Режим прерывания полезен и программисту, разрабатывающему программное обеспечение ЭС. В его распоряжение предоставляются средства анализа внутренних переменных системы, перекрестные ссылки между функциями программ и т. п. Подобная информация обеспечивается блоком прерывания и трассировки.

По окончании вывода решений ЭС пользователь имеет возможность обратиться к блоку анализа состояний ЭС. Для получения информации, каким путем, к каким решениям и почему пришла программа, он может использовать программы объяснений. На этапе отладки ЭС часто более важен не столько вопрос, почему ЭС получила именно такое решение, сколько вопрос, почему программа не нашла ожидаемого пользователем решения. Для этой цели в СПЭИС предусмотрены средства выделения тех продукций, левые части которых были оценены системой как ложные или неизвестные. Их анализ позволяет выявить недостатки БЗ.

Все блоки средств поддержки процесса построения ЭС имеют связь друг с другом, что позволяет при использовании одного режима обращаться к другому.

В заключение отметим, что СПЭИС является результатом обобщения опыта по построению ЭС в медицине и на сегодняшний день удовлетворяет запросы разработчиков. Однако систему нельзя считать универсальной.

15.2. Коммерческая гибридная система ЭКСПЕРТ

Б. С. Кирсанов, Э. В. Попов

15.2.1. Общее описание. Система ЭКСПЕРТ может быть классифицирована как коммерческое гибридное программное инструментальное средство (ИС) типа настраиваемой оболочки ЭС (см. гл. 13) [Изделие ЭКСПЕРТ]. ИС ЭКСПЕРТ применяется как при разработке конкретной ЭС, так и при ее использовании. В первом случае говорят о работе ИС ЭКСПЕРТ в режиме приобретения знаний, а во втором — в режиме решения задач (режиме консультации). ИС ЭКСПЕРТ обеспечивает при решении задач использование традиционных программ и баз данных, созданных с помощью системы ЛИНГВИСТ (см. § 3.2).

Разработка конкретной ЭС (приложения) на основе оболочки ЭКСПЕРТ осуществляется в соответствии с общей технологией разработки ЭС (см. рис. 13.2). Однако, так как ЭКСПЕРТ как оболочка содержит в готовом виде основные программные компоненты, реализация этапов формализации и выполнения существенно упрощается. На этих этапах необходимо определить структуру БЗ, наполнить ее и настроить параметры оболочки на особенности конкретной проблемной области.

Для ЭС, построенной на базе ИС ЭКСПЕРТ, решение задачи сводится к вычислению коэффициентов определенности утверждений и (или) значений числовых атрибутов, характеризующих состояния проблемной области, называемых в дальнейшем целями. В ходе решения задач допускается использование внешних программ, написанных на алгоритмических языках, и подзадач, решаемых другими ЭС, созданными на основе системы ЭКСПЕРТ.

Система ЭКСПЕРТ реализует все основные функции классической ЭС (см. гл. 13): 1) представление знаний; 2) приобретение знаний; 3) решение задач; 4) объяснение полученного результата и используемых знаний; 5) диалоговое взаимодействие с пользователем.

Система выполнена в виде двух программно независимых компонентов: средств приобретения знаний и средств видения консультации. Средства консультации могут поставляться отдельно. Средства приобретения знаний реализуют функции 1, 2 и 5, а средства консультации — функции 1, 3, 4 и 5.

Представление знаний о проблемной области осуществляется с помощью следующих конструкций:

утверждений о состоянии проблемной области, снабженных коэффициентами определенности (характеризуют степень уверенности в истинности утверждений);

символьных и числовых атрибутов, описывающих качественные и количественные характеристики сущностей проблемной области;

правил, описывающих взаимосвязи значений атрибутов и коэффициентов определенности;

сценариев, определяющих порядок решения задач, разбиение задачи на подзадачи, порядок выдачи сообщений и использования внешних программ.

Средства приобретения знаний реализуют жесткую схему инициализации диалогов, обеспечивающую полноту, корректность и сопряженность знаний, вводимых экспертом в систему. Вводимые утверждения, символьные и числовые атрибуты представляются в табличной форме. Правила могут быть представлены как в формализованном виде, так и в виде сообщений на естественном (русском) языке (ЕЯ). В последнем случае сообщения ЕЯ преобразуются лингвистическим транслятором в формализованное представление. Дополнительно средства приобретения знаний выполняют следующие функции: отображение введенных знаний на экран видеотерминала и на принтер в форматированном текстовом виде; тестирование и компиляцию знаний в сеть вывода с целью повышения скорости решения задач. В процессе компиляции из правил, атрибутов и утверждений строится сеть вывода, в явном виде содержащая все взаимодействия, обусловленные правилами.

Программные средства компонента приобретения знаний могут быть подразделены на интерактивные, которые взаимодействуют с экспертом и инженером по знаниям, и на утилиты, взаимодействующие только с программными средствами (рис. 15.1).

Интерактивные средства приобретения знаний выполняют следующие функции:

ввод, просмотр и корректировку утверждений, символьных и числовых атрибутов, используемых для описания состояний проблемной области и для решения задач;

ввод, просмотр и корректировку правил;

ввод, просмотр и корректировку описаний сценариев.

К моменту ввода правил все используемые в них переменные и утверждения должны быть описаны.

Утилиты средств приобретения знаний выполняют:

семантико-синтаксический контроль вводимых знаний и их преобразование из ЕЯ-формы в формализованную (данная функция реализуется программой, называемой лингвистическим транслятором);

преобразование переменных, утверждений и правил в сеть вывода (программа-компилятор);

вывод знаний в форматированном виде на принтер (программа-отображение).

Результатом работы средств приобретения знаний является:

внешнее представление знаний, используемое для отображения пользователю;

внутреннее представление знаний в виде сети вывода, используемое для решения задач.



Рис. 15.1. Средства приобретения знаний системы ЭКСПЕРТ

Структура компонента ведения консультации приведена на рис. 15.2. Управление ведением консультации осуществляет ДИАПОМ. В ходе консультации ДИАПОМ обеспечивает возможность одновременного поддержания по одному диалогу с каждым интерактивным средством. Интерактивные средства ведения консультации включают:

интерпретатор сети вывода, обеспечивающий решение задач на основе информации, полученной от пользователя, и вычислительной работы, определенной топологией сети вывода;

средства объяснения, поясняющие пользователю как процесс решения задачи, так и его результаты;

средства оказания помощи, инициируемые в случае затруднений у пользователя в ответах на вопросы системы.

Интерпретатор сценариев управляет процессом решения задачи в соответствии с обрабатываемым сценарием: задает исходные цели интерпретатору сети вывода и порядок просмотра вершин сети вывода; инициирует вызов подзадачи и внешних программ (для разрешения подцелей); определяет завершение задачи и выдаваемые данные.

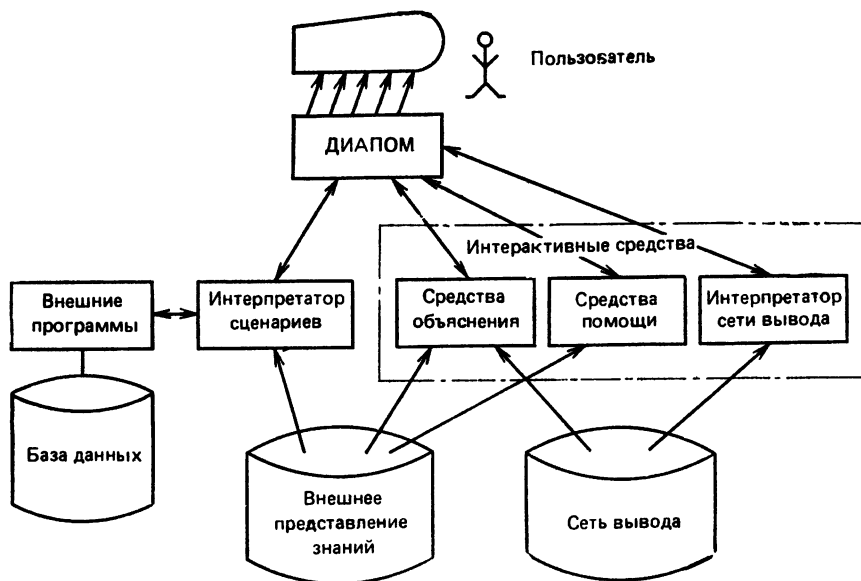


Рис. 15.2. Средства консультации системы ЭКСПЕРТ

Исходной информацией для работы компонента ведения консультации является сеть вывода и внешнее представление знаний. Сценарии решения задач не компилируются.

Вывод решения в системе осуществляется следующим образом. Для каждой исходной цели, заданной в сценарии, строится по сети вывода от целей перебором в глубину план ее достижения. План представляет собой цепочку правил, выполнение которых потенциально способно привести к достижению цели. При получении новых данных о проблемной области (из правил, от пользователя, от внешних программ или БД) цепочка правил, если условия всех правил окажутся истинными, последовательно выполняется, в противном случае пересматривается (строится новый план). Управление выводом (построением цепочек правил) в системе ЭКСПЕРТ осуществляется с помощью переупорядочивания правил или предложений сценария, содержащих метainформацию.

В ходе консультации пользователь может установить такой способ работы, при котором невыводимые цели получают значения по умолчанию (см. п. 15.2.2). Значения целей вычисляются один раз сразу всеми способами (числовые атрибуты — единственным способом) и не изменяются иначе, как по команде пользователя.

Средства консультации позволяют осуществлять повторное решение задачи со значениями некоторых целей, указанными пользователем, или с теми же исходными данными, но с измененной БЗ.

Средства объяснения обеспечивают доверие пользователя к полученному результату решения задачи и локализацию причин неудачи при решении задачи (что позволяет выявить ошибочные и недостающие знания и облегчить тестирование системы в режиме приобретения знаний), предоставляя по требованию пользователя следующую информацию: 1) зачем системе необходимы запрашиваемые у пользователя данные; 2) каковы значения вычисленного атрибута; 3) как получено значение коэффициента определенности или числового атрибута; 4) как изменится решение при изменении входных данных.

Средства объяснения обеспечивают также трассировку процесса решения задачи, в ходе которой объясняется каждое действие системы и используемая ею информация.

Средства диалогового взаимодействия обеспечивают взаимодействие с пользователем и в режиме приобретения знаний, и в режиме консультации. Эти средства поддерживают predetermined сценарием (см. п. 15.2.2) структуру диалога и осуществляют анализ, синтез и обработку текущих сообщений. Система может осуществлять одновременно несколько диалогов на различные, но связанные темы. Состояние каждого диалога отображается в виде окна на экране дисплея. Расположение окон определяется автоматически, однако пользователю предоставлены возможности по изменению расположения и размеров окон.

В работе с системой ЭКСПЕРТ применяются различные языковые средства. В соответствии с основными режимами работы в системе выделены языковые средства режимов приобретения знаний и консультации. Кроме того, в системе определен язык диалогового полиокононого монитора (ДИАПОМ), который может использоваться в обоих режимах.

Языковые средства режима приобретения знаний предназначены для представления знаний о проблемной области и для управления процессом их ввода в систему, языковые средства ведения консультации — для организации кооперативного решения задач (пользователем и ЭС) и управления процессом ведения консультации. Язык диалогового полиокононого монитора обеспечивает возможность параллельного ведения нескольких диалогов и переключения внимания между ними. Языковыми средствами приобретения знаний пользуется эксперт и инженер по знаниям. Пользователю достаточно знания языковых средств консультации. Знание языка ДИАПОМ не обязательно ни для эксперта, ни для пользователя. Однако использование этого языка предоставляет большие возможности участникам общения, особенно в режиме приобретения знаний, когда требуется параллельный просмотр и ввод в БЗ различной информации.

15.2.2. Языковые средства приобретения знаний. Языковые средства приобретения знаний реализованы в виде языка представления знаний (ЯПЗ), позволяющего описывать знания о проблемной области и управлять процессом накопления знаний (см. п. 15.2.3). Ввод знаний осуществляется на основе заранее разработанных диалогов с использованием шаблонов синтаксических конструкций ЯПЗ.

Процесс представления знаний (этап формализации) начинается с выявления и описания на ЯПЗ понятий и отношений проблемной области. В системе ЭКСПЕРТ они описываются с помощью атрибутов и утверждений.

Для выражения качественных характеристик проблемной области используются символьные атрибуты, а для количественных — числовые. Например, возраст пациента можно определить как числовой атрибут ВОЗРАСТ, значения которого изменяются в диапазоне от 0 до 150; диагноз — как символьный атрибут ДИАГНОЗ со следующими возможными значениями: абсцесс легкого, авитаминоз, агранулоцитоз, ...

Предложение вида «значение символьного атрибута А есть В» называется утверждением о состоянии предметной области. При решении неформализованных задач истинность или ложность многих утверждений не может быть установлена абсолютно точно, поэтому в ЭС используются числовые оценки коэффициента определенности утверждений, отражающие веру эксперта или системы в правдивость этих утверждений. В ЯПЗ коэффициент определенности (D) изменяется в диапазоне от -100 до $+100$, причем -100 обозначает ложно, от -100 до 0 — наверное ложно, 0 — неизвестно, от 0 до $+100$ — наверное истинно, $+100$ — истинно. Значением утверждения Н является его коэффициент определенности D. Таким образом, если утверждение Н ложно, то коэффициент его определенности $D(N)$ равен -100 ; если оно истинно, то $D(N) = 100$.

Для описания числового атрибута необходимо указать: имя атрибута; тип (числовой); текст комментария (не обязательно); диапазон допустимых значений; значение по умолчанию (не обязательно). Например:

АТРИБУТ «Возраст»

ТИП ЧИСЛОВОЙ

КОММЕНТАРИЙ: «Возраст пациента учитывается при ...»

ДИАПАЗОН: 0; 150

ПО УМОЛЧАНИЮ: —

Числовые атрибуты могут использоваться для описания понятий проблемной области, а также вспомогательных переменных в правилах.

Для описания символического атрибута необходимо указать: имя атрибута; тип (символьный); текст комментария (не обязательно); список значений. Список значений состоит из описания значений. Для описания значения символического атрибута указываются: значение; имя соответствующего утверждения (не обязательно); значение (определенность) утверждения по умолчанию (не обязательно); комментарий (не обязательно). Например:

АТРИБУТ «Диагноз»

ТИП СИМВОЛЬНЫЙ

КОММЕНТАРИЙ: «Постановка диагноза является основной целью консультации»

ЗНАЧЕНИЕ «Абсцесс легкого»

ИМЯ УТВЕРЖДЕНИЯ: «Абсцесс»

КОММЕНТАРИЙ: «Установлен абсцесс легкого»

ЗНАЧЕНИЕ УТВЕРЖДЕНИЯ ПО УМОЛЧАНИЮ: 0;

.....

Если значение утверждения по умолчанию не задано, оно принимается равным 0 (т. е. априори ничего не известно о правдоподобности утверждения). Имя утверждения применяется в правилах и в объяснениях как сокращение. Комментарии к атрибутам и символическим значениям используются при формировании объяснений.

Для анализа и вычисления значений характеристик понятий проблемной области в системе ЭКСПЕРТ используются правила. Правила состоят из левой части (условия применимости) и правой части (действия). Интерпретация правила такова: если условие применимости выполняется, то можно выполнить действие правила. В системе ЭКСПЕРТ приняты следующие виды правил: правила-вопросы, получающие информацию из внешних источников; арифметические; правила нечеткой логики (логические); байесовские.

Правила-вопросы позволяют в ходе консультации вводить в ЭС данные о конкретной ситуации проблемной области. Арифметические правила предназначены для вычисления значений числовых атрибутов, а также для получения значений коэффициентов определенности утверждений по нестандартным, предложенным экспертом формулам. Правила нечеткой логики позволяют выводить значения символических атрибутов с учетом неточности входной информации и (или) знаний. Байесовские правила позволяют определять коэффициент определенности тех утверждений, на истинности которых влияют разные факторы с различной значимостью.

Правила в общем случае имеют вид

[ЕСЛИ <логическое выражение>] ТО <текст процедуры> [<комментарий>]

Часть «ЕСЛИ ...» — условие применимости правила — не обязательна, часть «ТО ...» указывает в тексте процедуры, какие действия должна выполнять система для вычисления значения цели. Комментарии используются при формировании объяснений.

Правило может применяться, если значение его логического выражения больше 0. Логическое выражение строится с помощью операций конъюнкции

(&), дизъюнкции ($\vee$), отрицания ($\sim$) и скобок, при этом определенность логического выражения в каждом случае вычисляется следующим образом:

$$\begin{aligned}\&: D1\&D2 &= \min(D1, D2); \\ \vee: D1\vee D2 &= \max(D1, D2); \\ \sim: \sim D &= -D.\end{aligned}$$

где $D1$ и $D2$ — логические величины, т. е. величины, значения которых изменяются в интервале от -100 до $+100$.

В качестве логических величин могут использоваться: логические выражения (так как их значения являются логическими величинами); числовые параметры, значения которых интерпретируются пользователем (экспертом) в данной задаче как логические; коэффициенты определенности утверждений; результаты обращения к внешним программам, интерпретируемые как логические (например, программы, определяющие, обладают ли интересующие нас объекты некоторыми свойствами; результатами обращения к таким программам могут быть значения от -100 до $+100$); значения арифметических отношений $=$ (равно), $>$ (больше), $<$ (меньше).

На числовые атрибуты в формальных выражениях можно ссылаться по именам, на утверждения — по именам утверждений или с помощью пар вида

<имя символического атрибута> · <имя значения>

Приведем примеры логических выражений, представленных в формальном виде (комментарий дан в квадратных скобках):

$45\&(5>3)$ — [дает значение $+45$];

$(\text{возраст} > 40)\& \text{профессия.шахтер}$ — [дает значение -100 при $\text{возраст} = 25$ (числовой атрибут), $\text{профессия} = \text{шахтер}$ (символьный атрибут)].

Эти же выражения на ЕЯ имеют вид

45 и 5 больше 3;

возраст больше 40 лет и профессия шахтер.

У всех видов правил структура условия применимости одинакова. Разные виды правил различаются текстом процедуры в их правой части.

Правила-вопросы делятся на простые и сложные (общие вопросы). Простые вопросы позволяют получать значения числовых атрибутов или коэффициентов определенности отдельных утверждений, а сложные вопросы — вводить значение символического атрибута, т. е. коэффициенты определенности сразу нескольких утверждений.

Простые правила-вопросы имеют вид

[ЕСЛИ <логическое выражение> ТО] A=ВОПРОС
[: <текст вопроса>] [<комментарий>]

где A — имя числового атрибута или имя утверждения.

Если текст вопроса не указан, система использует стандартный текст: для числового атрибута «Какова величина A?»; для утверждения «Насколько Вы уверены в том, что A?»

Задавая вопрос, система подсказывает, в каком виде она ожидает получить ответ.

Отметим, что в связи с тем, что коэффициент определенности имеет логарифмическую шкалу, то для общения с конечным пользователем используется более удобная для него линейная шкала оценок определенности в диапазоне от -5 до $+5$, в которой система и будет ожидать ответов об утверждениях.

При сложных вопросах правила имеют вид

[ЕСЛИ <логическое выражение>] ТО (<ИМЯ символического атрибута> =
ВОПРОС [<Текст вопроса>] [ИЛИ] [<КОММЕНТАРИЙ>])

В зависимости от того, указан или нет параметр ИЛИ, сложный вопрос задается в одной из двух форм. Если ИЛИ указано, то считается, что поль-

зователь во время консультации может либо ничего не знать о значении данного символьного атрибута, либо точно знать, какое из его значений имеет место. В этом случае система задает вопрос (если текст вопроса не указан, используется стандартный текст «Укажите, какой из перечисленных вариантов имеет место для <имя символьного атрибута>»), а затем выводит список перенумерованных возможных значений атрибута. В качестве ответа ожидается номер выбранного варианта или 0 (неизвестно). Если введен номер варианта, значение соответствующего утверждения принимается равным +100, а остальных утверждений о данном символьном атрибуте — равными —100. Если введен 0, все утверждения считаются неизвестными.

Если ИЛИ не указано, то считается, что пользователь не может во время консультации точно указать единственное значение символьного атрибута; ожидается, что он укажет те значения, которые возможны, и при этом отметит, насколько он уверен в каждом из них. Система задает вопрос (стандартный текст: «Насколько Вы уверены в том, что <имя символьного атрибута>»), а затем перечисляет возможные значения символьного атрибута. В этом списке пользователь должен проставить коэффициент определенности у тех значений, о которых он что-то знает.

Арифметические правила имеют вид

[ЕСЛИ <логическое выражение> ТО] A=<арифметическое выражение>
[<комментарий>]

где A — имя числового атрибута, имя утверждения или ссылка на утверждение вида <имя символьного атрибута>.<имя значения>.

Арифметические выражения строятся с помощью арифметических операций (+, —, *, /) и функций. Можно использовать как стандартные функции, так и обращения к прикладным программам. Введены следующие стандартные арифметические функции: $\exp(X, Y)$ — число X в степени Y; $\text{sq}(X)$ — квадратный корень из X; $\lg(X)$ — десятичный логарифм X; $\text{abs}(X)$ — абсолютное значение X; $\text{pb}(X)$ — вероятность для коэффициента определенности X; $\text{dx}(X)$ — коэффициент определенности для вероятности X и др. Примеры арифметических правил:

- 1) ЕСЛИ $\text{abs}(\text{надежность.высокая}) < 5$
ТО $\text{информативность} - \text{консультации} = 0$
- 2) ЕСЛИ $(\text{вес} > 90) \& (\text{рост} < 170)$
ТО $\text{доза} - \text{препарата} = 0.1 * \text{вес} - 0.01 * (\text{норм} - \text{вес/рост})$

Правила нечетной логики (логические) имеют вид

ЕСЛИ [<логическое выражение> ТО A [($\<\text{вес правила}>$)] [<комментарий>]
где A — имя числового атрибута, имя утверждения, или выражение <имя символьного атрибута>.<значение>; <вес правила> = число от 0.00 до 1.00. При выполнении правила нечеткой логики вычисляется логическое выражение, и если оно положительно, то произведение его значения и веса правила присваивается A. Если вес правила не указан, то он принимается равным 1.

Примеры правил нечеткой логики:

- 1) ЕСЛИ $\text{стаж.большой} \vee \text{должность.вед.спец.}$
ТО $\text{квалификация высокая} (0.9)$
- 2) ЕСЛИ $(\text{возраст} < 15) \vee \text{аллергия}$
ТО противопоказания
- 3) ЕСЛИ $\text{надежность} \& (\sim \text{нагрузка})$
ТО $\text{фактор} - \text{устойчивости} (0.8)$

Байесовские правила применяются для вычисления коэффициентов определенности тех утверждений, об истинности которых можно судить по выполнению ряда факторов (симптомов), имеющих разную значимость. Байесовское правило задается в виде таблицы (табл. 15.1). В первом столбце перечисляются утверждения-факторы, знания об истинности или ложности которых влияют

Структура байесовского правила

ЕСЛИ (лог. выражение) ТО D (G) ОПРЕДЕЛЯЕТСЯ ПО:

ФАКТОР	A_i	B_i
Утверждение 1	A_1	B_1
Утверждение 2	A_2	B_2
. . .	. . .	. . .
Утверждение M	A_M	B_M

на коэффициент определенности целевого утверждения, во втором и третьем столбцах указываются подтверждающие и опровергающие веса факторов, т. е. степень их влияния на целевое утверждение. Эти веса можно задавать как числовыми атрибутами, так и числовыми константами. Если подтверждающий (опровергающий) вес фактора 0, то это свидетельствует о незначимости фактора. Чем больше значимость фактора, тем больше должны быть его подтверждающий (опровергающий) веса. Абсолютное значение суммы положительных (отрицательных) весов не должно превышать 100.

При выполнении действия байесовского правила коэффициенту определенности целевого утверждения сначала присваивается значение по умолчанию (априорное). Затем выбирается первая строка таблицы. Если коэффициент определенности утверждения-фактора 0, то значение определенности целевого утверждения не изменяется. Если коэффициент определенности 100, то к вычисляемой определенности добавляется подтверждающий вес, а если —100, то добавляется опровергающий вес. Если коэффициент определенности фактора лежит в интервале от 0 до 100 (—100 до 0), к текущему значению вычисляемого коэффициента определенности прибавляется результат интерполяции подтверждающего (опровергающего) веса, определяемый по приводимым ниже формулам. После этого аналогично рассматриваются все последующие строки таблицы.

Таким образом, вычисление коэффициента определенности целевого утверждения G байесовского правила осуществляется по формуле

$$D(G/H_1, H_2, \dots, H_M) = D_{\text{апр}}(G) + I(H_1) + I(H_2) + \dots + I(H_M),$$

где $H_1, H_2, \dots, H_M$ — факторы; $D_{\text{апр}}(G)$ — априорное значение коэффициента определенности целевого утверждения; $I(H_i)$ — интерполяция подтверждающего или опровергающего веса фактора H_i .

Интерполяция веса вычисляется по формулам

$$I(H_i) = \begin{cases} 33 \lg \left(1 + \frac{P(H_i) - P_{\text{апр}}(H_i)}{1 - P_{\text{апр}}(H_i)} (10^{A_i/33} - 1) \right) & \text{при } D(H_i) > 0, \\ 33 \lg \left(10^{B_i/33} \left(1 - \frac{P(H_i)}{P_{\text{апр}}(H_i)} \right) + \frac{P(H_i)}{P_{\text{апр}}(H_i)} \right) & \text{при } D(H_i) < 0, \end{cases}$$

где A_i и B_i — подтверждающий и опровергающий веса фактора H_i ; $P = 10^{D/33} / (1 + 10^{D/33})$.

Пример байесовского правила

остеомиелит челюстей ОПРЕДЕЛЯЕТСЯ ПО:

ФАКТОР	A_i	B_i
Сильные боли	4	—2
Сопутствующие заболевания	Тяжесть	0
Симптом Венсана	20	—30
Обширное воспаление	15	—20

Пример правила, вычисляющего коэффициент определенности утверждения «остеомиелит челюстей», приведен в табл. 15.2. Покажем, как в этом случае будут проводиться вычисления.

Пусть априорный коэффициент определенности утверждения «остеомиелит челюстей» 0, а выведенное в ходе консультации значение числового атрибута «тяжесть» 5. Пусть также программе удалось усановить:

$$D(\text{сильные боли}) = 30;$$

$$D(\text{сопутствующие заболевания}) = 100;$$

$$D(\text{симптом Венсана}) = 0;$$

$$D(\text{обширное воспаление}) = -100.$$

Предположим, что в модели указан априорный коэффициент определенности утверждения «сильные боли» 20,00. В этом случае по указанным выше формулам получим

$$P_{\text{апр}}(\text{сильные боли}) = \frac{10^{20/33}}{1 + 10^{20/33}} = 0,80;$$

$$P(\text{сильные боли}) = \frac{10^{30/33}}{1 + 10^{30/33}} = 0,89.$$

Вычисляя значение целевого утверждения, программа определит:

$$I(\text{сильные боли}) = 33 \lg \left(1 + \frac{0,89 - 0,80}{1 - 0,80} (10^{4/33} - 1) \right) = 1,93;$$

$$I(\text{сопутствующие заболевания}) = 5,00$$

$$I(\text{симптом Венсана}) = 0,00;$$

$$I(\text{обширное воспаление}) = -20,00.$$

Итак,

$$D(\text{остеомиелит челюстей}) = 0,00 + 1,93 + 5,00 + 0,00 - 20,00 = -13,07.$$

Логические и байесовские правила отличаются не только способом вычисления определенности, но и способом объединения определенностей, если коэффициент определенности некоторого утверждения вычисляется несколькими

правилами. Логические правила при объединении определенностей берут максимум из коэффициентов определенностей объединяемых утверждений, а байесовские правила вычисляют коэффициент определенности по приведенным выше формулам, учитывающим вклад в результат коэффициентов определенности всех утверждений. Утверждения, атрибуты и правила позволяют определять понятия проблемной области и способы вычисления их значений, но они не содержат никакой информации о порядке проведения консультации: какие атрибуты являются целевыми, какие сообщения необходимо выдавать на экран дисплея в различных ситуациях и т. д. Эта информация задается в сценарии консультации. Сценарий представляет собой последовательность операторов (команд), каждый из которых может иметь метку. Так же, как и правила, операторы могут иметь условия применимости «ЕСЛИ <логическое выражение> ТО».

Общий вид оператора сценария:

[<метка>] : [ЕСЛИ <логическое выражение> ТО] <действие>

где «действие» является одной из следующих команд:

ЦЕЛЬ $A_1, A_2, \dots, A_k$ вычислить значения A_i (числового атрибута, утверждения или символьного атрибута)

СБРОС отменить все выведенные системой значения, а также ответы на заданные во время консультации вопросы (включая другие данные, полученные из различных внешних источников)

СБРОС ВЫВОДА отменить все выведенные значения, кроме ответов на вопросы и данных, полученных из внешних источников

СБРОС $A_1, \dots, A_n$ сбросить значения целей $A_1, \dots, A_n$
ПЕРЕХОД <метка> перейти к оператору с заданной меткой

СООБЩЕНИЕ
 (<список текстов и целей>) выдать на экран сообщение, образованное из списка текстов с подставленными в них выведенными значениями целей

ВН_ПОДЗАДАЧА
 <имя подзадачи>
 (<список целей>) определить внешнюю подзадачу,

ВОЙТИ <имя подзадачи> (<список целей>) решить внутреннюю или внешнюю подзадачу

ПАРАМЕТРЫ $A_1, \dots, A_k$ определить внешние параметры подзадачи

СТОП закончить работу.

Описание задачи в сценарии может состоять из описаний подзадач в виде модулей. Каждый модуль является независимым фрагментом описания задачи и предназначен для вычисления некоторых целей. Например, можно оформить в виде модулей различные поддиалоги консультации. Некоторые модули могут применяться в различных задачах. Модули, описанные в одной задаче и используемые только этой задачей, называются внутренними. Модули, применяемые в разных задачах, описываются отдельно; они называются внешними. Если в задаче предполагается существование внешней подзадачи, то она определяется соответствующим действием в сценарии. Обращение к этой подзадаче происходит в том случае, когда появляется необходимость в значении какой-либо цели, упомянутой в списке целей действия **ВН_ПОДЗАДАЧА**. Действие **ВОЙТИ**

позволяет обратиться к внутренней или внешней подзадаче из сценария консультации.

Каждая подзадача выполняется как конкретная ЭС, решающая определенные задачи. Чтобы обеспечить ее вызов из других задач, вводится ее описание в сценарий вызываемой задачи. Для внутренней задачи

ЗАДАЧА <имя задачи> [:<текст комментария>]

для внешней

ВН-ЗАДАЧА <имя задачи> [:<текст комментария>]

Формальные параметры задачи определяются действием ПАРАМЕТРЫ.

Для системы ЭКСПЕРТ с использованием упомянутых выше команд разработаны стандартные интерфейсные модули, обеспечивающие получение ЭС исходных и промежуточных данных от информационных систем, созданных с помощью системы ЛИНГВИСТ (см. § 3.2).

15.2.3. Управление накоплением знаний. Для обеспечения ввода знаний в системе ЭКСПЕРТ реализован ряд диалогов по вводу информации. Управление процессом ввода информации осуществляется с помощью специальных команд. Выделены команды технологические и связанные с вводом конкретных знаний. К технологическим относятся следующие: загрузка задачи с внешних носителей и подготовка знаний к корректировке; восстановление задачи (игнорируются все изменения, проведенные пользователем); тестирование и компиляция задачи; сохранение задачи (изменения, проведенные пользователем, сбрасываются на внешний носитель во внешнем представлении в откомпилированном виде), после этой команды происходят необратимые изменения БЗ.

Команды, связанные с вводом конкретных знаний, обеспечивают диалог по вводу знаний, их просмотр и корректировку. По виду вводимых знаний выделяют команды ввода утверждений, символьных атрибутов, числовых атрибутов, сценариев, комментариев и правил.

Первой всегда должна выполняться команда загрузки задачи. После того как задача загружена, можно выполнять любую команду, кроме команды ввода правил. Команду ввода правил можно выполнять только в момент ввода утверждения, значения символьного или числового атрибута. В каждом из этих случаев доступными для просмотра, ввода, корректировки являются только те правила, которые осуществляют вычисление соответствующих значений. Исключение составляют правила, имеющие в правой части общий вопрос, который позволяет найти коэффициент определенности любого утверждения, полученного из символьного атрибута. Инициализация подобного правила осуществляется из диалога по вводу описания символьного атрибута.

Все команды могут привести к сложному диалогу со значительным количеством поддиалогов, ориентированных на обеспечение полноты и корректности вводимых знаний. Инициализация диалогов осуществляется системой. Пользователь может перехватить инициативу с помощью команд ДИАПОМа.

15.2.4. Средства консультации в системе ЭКСПЕРТ. Ведение консультации осуществляется в соответствии с жестко встроенной глобальной структурой диалога, определяющей последовательность поддиалогов и предписанные способы перехода от одного поддиалога к другому. В рамках глобальной структуры инициализация поддиалогов осуществляется либо автоматически, либо по командам пользователя.

Первой командой пользователя должна быть команда «Начать консультацию». По данной команде начинается тематический диалог. Среди поддиалогов, предусмотренных в глобальной структуре, основную роль играет тематический поддиалог, поддиалоги объяснения и помощи. Поддиалоги объяснения и помощи ориентированы на обеспечение доверия пользователя к полученному результату и оказание ему помощи в затруднительных случаях.

Структура тематического поддиалога определяет, какие задачи решает пользователь, а какие — ЭС. Структура этого поддиалога не является жесткой, а задается сценарием консультации (см. п. 15.2.2) и правилами БЗ, сформиро-

ванной экспертом и инженером по знаниям. Сценарий и правила определяют порядок задания вопросов пользователю и решения задач (подзадач) системой. В системе ЭКСПЕРТ используются фиксированные формы шагов тематического поддиалога (см. в п. 15.2.2 правила-вопросы); конкретные вопросы, требующие оценить определенность утверждения; общие вопросы, требующие установить значение силового атрибута или оценить определенность всех утверждений, полученных из символического атрибута; общие вопросы, требующие выбрать из набора альтернатив одно значение. Пользователь может в качестве реакции на вопрос системы выбрать следующие действия: дать ответ; отказаться от ответа; в соответствии с глобальной структурой диалога выдать команду, инициирующую новый диалог.

Для выбора различных течений консультации, предусмотренных глобальной структурой диалога, в системе ЭКСПЕРТ реализованы команды, позволяющие: завершить диалог; оказать помощь пользователю в формировании ответов системе; отобразить результаты работы систем и ответы пользователя; изменить результаты ответов пользователя или выведенные системой значения; повторить консультацию; объяснить процесс функционирования системы.

Возможности поддиалогов объяснения предполагают ответы на следующие типы вопросов пользователя: зачем системе потребовалась запрашиваемая информация; как получено указанное значение; как повлияет изменение указанного значения на конечный результат.

Глобальная структура диалога позволяет рекурсивно вызывать поддиалоги (кроме тематического) в рамках заданной структуры. Пользователь может осуществлять движение по глобальной структуре диалога с помощью команд, инициирующих и завершающих диалог.

15.2.5. Дополнительные языковые средства. Чтобы расширить возможности пользователя, ему предоставлен язык ДИАПОМ. Этот язык позволяет пользователю переходить от диалога к диалогу, не завершая их, а лишь откладывая и возобновляя по требованию пользователя. Разрешается одновременно вести несколько диалогов и управлять отображением состояния диалогов на экране дисплея, представляемых в виде окон. Позволяется, например, при возникновении в основном диалоге некоторой непонятной ситуации создать (не завершая основной диалог) один или несколько поддиалогов, помогающих устранить непонимание.

К командам управления диалогами (поддиалогами) относятся: инициализация; активация; завершение; просмотр.

Инициализация диалога осуществляется выдачей команд языка режима консультации или приобретения знаний из текущего диалога. ДИАПОМ позволяет инициализировать новые диалоги, не завершая старые.

Просмотр диалогов делается с целью их активации или анализа их состояния.

В рамках языка ДИАПОМ реализованы команды просмотра диалогов как по логическим связям, так и по расположению окон на экране дисплея. Просмотр диалогов по логическим связям позволяет осуществить: переход к диалогу, породившему процесс, на котором сконцентрировано внимание; переход к диалогу, инициированному из текущего диалога, и т. п. Просмотр диалогов по расположению окон на экране дисплея обеспечивает перенос внимания только на окна, закрытые не полностью. При этом внимание переносится на диалог, представляемый окном, ближайшим к текущему окну сверху, снизу, справа или слева.

Активация диалога подготавливает к работе диалог ранее инициализированный, но незавершенный. При завершении активного диалога активируется диалог, из которого был порожден завершившийся диалог.

15.2.6. Реализация системы. ЭКСПЕРТ работает на ЭВМ типа ЕС-1841 или IBM PC/XT,AT соответственно с операционной системой АЛЬФА DOS (DOS-16) издания 3.0 и выше, MS DOS версии 3.1 и выше. Использование всех функциональных возможностей системы требует конфигурации ЭВМ с емкостью оперативной памяти не менее 512 Кбайт, принтером, цветным графическим дис-

плеем и двумя накопителями на гибких магнитных дисках емкостью не менее 360 Кбайт. Система реализована на языке программирования Сн, объем программного обеспечения около 360 Кбайт. Автономный решатель системы, работающий без диалога на основе полученных при вызове данных (как подпрограмма), требует для работы 60 Кбайт, а для работы совместно с программой ЛИНГВИСТ — 512 Кбайт.

С помощью оболочки ЭКСПЕРТ можно создавать ЭС, решающие задачи в проблемных областях, содержащих 450 утверждений и числовых атрибутов и до 1000 правил в каждой внешней задаче.

Первая версия системы ЭКСПЕРТ передана для распространения НПО «Алгоритм». В 1988 г. завершены работы над второй версией системы, обеспечивающей общение с пользователем в виде изображений. Дополнительно о развитии системы ЭКСПЕРТ см. п. 16.4.3.

15.3. Коммерческая система ФИАКР

С. Ю. Соловьев, Г. Н. Соловьева

15.3.1. Общее описание. Система ФИАКР предназначена для построения ЭС, поддерживающих процесс принятия решений в задачах диагностики. В соответствии с типологией ЭС система ФИАКР относится к оболочкам ЭС.

Действующими лицами в сеансе консультации выступают пользователь, желающий получить заключение о некотором объекте исследования, и ЭС. Система целенаправленно задает ему вопросы о доступных характеристиках объекта, оформляя каждый свой вопрос в виде меню значений некоторого наблюдаемого признака. Задача пользователя — как можно точнее отвечать на вопросы системы. Получив очередную порцию информации, ЭС пытается построить искомым заключение. Если же информации оказывается недостаточно, система обращается к пользователю с новыми вопросами. Консультация заканчивается сразу по получении заключения.

Система ФИАКР разрабатывалась на основе предварительного анализа задачи диагностики растений. В дальнейшем именно эта проблемная область была использована в качестве модельной задачи для проверки реальных возможностей системы. На содержательном уровне задача ЭС ФитоАктиваторов Растений на примере Томатов (ФИАКР-Т) состоит в том, чтобы по морфологическому описанию томата определить характерные для него типы соотношений между основными фитогормонами. Диагнозы такого рода используются на этапе планирования экспериментов по клеточной инженерии.

Система ФИАКР применима для проблемных областей, где описание внешних признаков, промежуточных суждений и целевых состояний может быть представлено в виде совокупности атрибутов, принимающих конечное число взаимоисключающих значений. Другими словами, на каждом объекте проблемной области атрибуты, участвующие в его описании, принимают ровно одно значение.

Атрибуты и их значения используются для описания декларативных знаний. Представление в системе ФИАКР процедурных знаний осуществляется путем отождествления отдельных атрибутов с именами программ. Когда в ходе консультации система обрабатывает значение такого атрибута, диалог с пользователем временно прерывается и автоматически вызывается указанная программа, причем полученное значение передается в программу в качестве аргумента.

В реальных задачах описание проблемной области содержит сотни различных атрибутов, что затрудняет работу эксперта. В связи с этим множество атрибутов разделяется на непересекающиеся классы, которые называются подсистемами. Например, подсистема ЛИСТ может включать атрибуты ОКРАСКА, РАЗМЕР, РАССЕЧЕННОСТЬ и т. д. Применение подсистем облегчает, в частности, выбор имен атрибутов. Так, в базе знаний (БЗ) могут одновременно присутствовать три различных атрибута ОКРАСКА, принадлежащие подсистемам СТЕБЕЛЬ, ЛИСТ и ПЛОД. Вместе с тем для атрибутов, характеризую-

щих целевые состояния, необходимо использовать уникальные имена. Признаковые модели указанного вида широко распространены в таких описательных науках, как биология, медицина и т. п. В связи с этим логическая организация БЗ в системе ФИАКР без труда воспринимается экспертами. Однако для инженера по знаниям необходимо привести более точное описание БЗ.

Тройка (<ПОДСИСТЕМА>, <АТРИБУТ>, <ЗНАЧЕНИЕ>) называется дескриптором свойства и записывается в виде <ПОДСИСТЕМА>—<АТРИБУТ>: <ЗНАЧЕНИЕ>. Например, ЛИСТ_ОКРАСКА:ЗЕЛЕНЫЙ. Дескрипторы свойств являются частным случаем утверждений, представленных в БЗ.

В ходе консультации каждое утверждение характеризуется состоянием — значением одноименного слова, приписанного утверждению. В ходе консультации значения слотов могут изменяться. Например, четыре характеристики утверждений, соответствующих четырем значениям слота СОСТОЯНИЕ: установленные, опровергнутые, неопределенные и конфликтные. Перед началом консультации большинство утверждений, включая все дескрипторы свойств, находятся в неопределенном состоянии, остальные считаются установленными априори. По мере получения информации об объекте исследования, утверждения могут опровергаться или устанавливаться. Случай конфликтного состояния отражает наличие в БЗ двух независимых выводов, один из которых опровергает данное утверждение, а другой — устанавливает. (Ситуации такого рода возникают, как правило, при наличии противоречий в исходных данных.)

В системе ФИАКР БЗ формируется путем накопления знаний отдельных модулей. В каждом модуле представляется ограниченное множество утверждений, для которых задаются правила вычисления состояний. В системе предусматривается пять стандартных модулей: альтернатива, несовместимость, запрет, продукция, модуль импликации.

Модуль альтернативы либо опровергает все оставшиеся утверждения при наличии одного установленного, либо устанавливает утверждение, если все остальные были опровергнуты. Типичным примером альтернативы является описание атрибута. В системе ФИАКР все атрибуты в ходе консультации функционируют как модули знаний. Модуль альтернативы задает полный ряд парно несовместимых свойств, из которых для каждого объекта консультации имеет место ровно одно.

Модуль несовместимости позволяет задавать неполные ряды несовместимых свойств, для них возможен только первый вид функционирования модуля альтернативы, т. е. позволяет только опровергать входящие в него утверждения.

Модуль запрета [Закревский, 1982] опровергает одно из составляющих его утверждений, если установлены все остальные, позволяет задавать комбинации дескрипторов свойств, которые в таком сочетании никогда не встречаются.

Последние два из перечисленных модулей реализуют различные типы зависимостей вида «ЕСЛИ ..., ТО ...», позволяя устанавливать следствия при наличии всех посылок. В случае опровергнутого следствия начинает функционировать модуль импликации как запрет комбинации своих посылок. Для модуля продукции такое функционирование не предусмотрено. Обычно модуль импликации используется для задания в БЗ причинно-следственных связей, имеющих характер закона, а модуль продукции предназначен главным образом для выражения однонаправленных и зачастую субъективных указаний.

Дескрипторы свойств непосредственно связаны с формальной моделью внешнего мира. Вместе с тем в БЗ необходимо иметь аппарат для организации работы самой ЭС в ходе консультации. С этой целью в БЗ системы ФИАКР создаются особые утверждения, называемые флагами.

Во многих приложениях необходимо считаться с тем, что для отдельных объектов внешнего мира определенные атрибуты могут отсутствовать. Традиционный выход из этой ситуации состоит в том, что необязательному атрибуту приписывается специальное значение, которое интерпретируется как признак отсутствия атрибута вообще. В системе ФИАКР используется другой метод, в соответствии с которым каждому атрибуту в БЗ автоматически приписывается флаг существования вида БЛОК—<ПОДСИСТЕМА>= <АТРИБУТ>. Термин

БЛОК зарезервирован в системе так же, как и термины **БАЗА**, **АНКЕТА**, **ЦЕЛЬ** и **ШЕФ**. В ходе консультации система оперирует только теми дескрипторами свойств, для которых установлены флаги существования атрибутов.

Усеченный вариант реализации некоторых объектов зачастую объясняется отсутствием у них целых подсистем. Для отображения зависимостей на уровне структуры объектов в БЗ создаются флаги существования подсистем вида **БАЗА_БЛОК**=<**ПОДСИСТЕМА**>. Аналогично атрибутам подсистема в ходе логического вывода начинает функционировать как специфический модуль знаний, который устанавливает флаг существования подсистемы тогда и только тогда, когда установлены флаги существования всех ее атрибутов.

На каждом этапе консультации системе известно, какие именно атрибуты гарантированно существуют у данного объекта, хотя система может и не знать их точных значений. Совокупность этих атрибутов образует так называемую обязательную структуру объекта, исходя из которой ЭС задает вопросы и строит цепочки логического вывода. Обязательная структура задается установленными флагами существования атрибутов. В момент инициализации диалога флаги существования в БЗ устанавливаются только для априорно обязательных атрибутов, а для остальных или устанавливаются, или опровергаются в ходе консультации. Чтобы обеспечить эту возможность, инженер по знаниям должен выделить в формальной модели специальные признаки для уточнения структуры объекта и занести в БЗ модули, связывающие эти признаки с флагами существования. Таким образом, из всей совокупности модулей знаний выделяется отдельная группа — модули уточнения структуры объекта. В принципе кластеризацию модулей имеет смысл применять и для оригинальных экспертных знаний, которые можно разделять на группы в соответствии с оценками достоверности или по признаку авторства.

В связи с этим множество модулей, представимых в БЗ системы **ФИАКР**, разделяется на определенные группы, которые называют экспертными. Экспертная группа характеризуется именем и флагом активизации вида **БАЗА_ШЕФ**=<**ИМЯ_ГРУППЫ**>. В общем случае использование модуля в ходе логического вывода поставлено в двойную зависимость: от флага активизации его группы и от флагов существования атрибутов, соответствующих дескрипторам свойств, входящих в этот модуль.

Флаги активизации экспертных групп устанавливаются модулями знаний. Те же группы, которые являются обязательными для конкретной проблемной области, могут быть объявлены априорно обязательными; и их флаги активизации устанавливаются автоматически в момент инициализации диалога. Кроме того, в ходе диалога пользователь имеет право подключить к консультации новые группы модулей.

Несмотря на единообразное представление атрибутов в БЗ, их роль в ходе консультации может быть различной. Часть атрибутов используется в качестве вопросов пользователю. Эти атрибуты представляют собой наблюдаемые или измеряемые признаки объекта консультации. Значения остальных атрибутов определяются в ходе логического вывода. Часть выводимых атрибутов объявляется в системе как потенциально возможные цели консультации, такие атрибуты называют целевыми. Вообще говоря, в ходе консультации пользователя могут интересовать значения только некоторых целевых атрибутов. Это должно учитываться при организации диалога. Поэтому для целевых атрибутов в БЗ создаются так называемые флаги целей вида **БАЗА_ЦЕЛЬ**=<**АТРИБУТ**>. При инициализации диалога с пользователем система предлагает уточнить цели консультации. В этот момент в БЗ устанавливаются флаги целей для тех атрибутов, которые интересуют пользователя в данной консультации.

Для практического использования ЭС проблема организации диалога не менее важна, чем проблема адекватного представления знаний эксперта. В самом деле, если БЗ включает более 100 наблюдаемых признаков и при опросе используется раз и навсегда заложенный порядок вопросов, то работа с такой системой превратится для пользователя в тяжкую повинность. Поэтому в ЭС необходимо иметь средства для организации гибкого диалога.

В системе ФИАКР диалог с пользователем основан на механизме смены анкет. Анкета представляет собой группу наблюдаемых признаков, причем каждой из них ставится в соответствие флаг опроса вида: `БАЗА_АНКЕТА = <ИМЯ_АНКЕТЫ>`. Подчеркнем, что анкеты могут содержать признаки из разных подсистем. Средства управления диалогом разделяются на две части. К первой относятся флаги опроса и модули знаний, изменяющие их состояние, а вторую часть составляет процедура выбора очередной анкеты, заложенная в программы системы ФИАКР. В общем случае, получив информацию по всем признакам, перечисленным в анкете, система просматривает сначала список еще не использованных анкет. Для продолжения диалога выбирается первая по порядку анкеты с установленным флагом опроса.

Флаги опроса устанавливаются модулями знаний, отвечающими за управление диалогом. Эти модули формируются инженером по знаниям на этапе создания ЭС. Модули управления помимо дескрипторов свойств могут использовать флаги целей и флаги существования.

С содержательной точки зрения модули управления диалогом отражают процесс выдвижения гипотез о состоянии объекта консультации, хотя окончательное заключение системы формируют другие модули — модули проблемного наполнения БЗ. Часть флагов опроса может быть установлена в БЗ заранее, еще до начала консультации. Как правило, эта возможность используется для выбора начальной или стартовой анкеты опроса, информация из которой должна в дальнейшем активизировать работу ЭС по выводу заключения и управлению диалогом.

Система ФИАКР реализована на языке Паскаль, причем все машинно-независимые операторы вынесены в отдельные процедуры, что обеспечивает переносимость системы на мини-ЭВМ разных типов [Соловьева, 1985; Соловьев и др., 1985]. Разработка и отладка основной версии системы ФИАКР велась на ЭВМ СМ-4. Различные блоки программного обеспечения реализованы в виде отдельных программ. Взаимодействие программ при работе с БЗ обеспечивается командными файлами.

15.3.2. Режим приобретения знаний. Основной способ построения ЭС состоит в том, что знания о проблемной области записываются на языке представления знаний (ЯПЗ) системы ФИАКР и далее транслятором загружаются в БЗ. Базовыми понятиями языка являются синтаксические конструкции утверждений. Для записи модулей знаний применяется позиционный принцип интерпретации терминов. Описание БЗ на ЯПЗ представляет собой линейную последовательность модулей. В качестве примера рассмотрим описание одного из модулей управления диалогом

ВОМ ПРОДУКЦИЯ П.1 ДИАЛОГ

ЛИСТ _ ОКРАСКА : ЗЕЛЕНый

БАЗА _ БЛОК = ЛИСТ

БАЗА_АНКЕТА = ГИПОТЕЗА-А ЕОМ

Пара зарезервированных слов ВОМ и ЕОМ определяет соответственно начало и конец модуля. В первой строке задаются последовательно тип (ПРОДУКЦИЯ), имя (П.1) и экспертная группа (ДИАЛОГ) данного модуля. Вторая и третья строки определяют послыки продукции, а в последней строке содержится заключение. Горизонтальная черта позволяет разделять послыки и следствия.

Полное описание языка приводится в [Соловьев и др., 1986]. Как показывает опыт разработки конкретных ЭС, тексты на ЯПЗ без труда понимаются экспертами, однако запись содержимого БЗ с соблюдением всех формальных правил лучше всего возложить на инженера по знаниям.

Процесс отладки БЗ требует оперативной корректировки модулей. С этой целью в состав системы включены диалоговые редакторы, которые с формаль-

ной точки зрения позволяют создавать БЗ и без использования ЯПЗ. В действительности одних только диалоговых возможностей для построения реальных БЗ недостаточно. Через какой-то промежуток времени с начала заполнения БЗ в интерактивном режиме эксперт начинает испытывать определенные затруднения, связанные с невозможностью охватить и оценить достаточно большие фрагменты базы. В конце концов эксперт может просто потерять интерес к работе с системой. Таким образом, диалоговые редакторы, хотя и представляют собой необходимое средство поддержки режима приобретения знаний, не могут полностью компенсировать преимущества использования ЯПЗ.

Редактор модели проблемной области обеспечивает модификацию подсистем, атрибутов и их значений, а также позволяет изменять ролевые реквизиты атрибутов:

- тип шкалы значений (порядковая или номинальная);
- отношение атрибута к априорной структуре объекта (обязательный или необязательный);

- роль атрибута в ходе логического вывода (наблюдаемый или выводимый);
- возможность использования атрибута в качестве цели (нецелевой или целевой);

- тип атрибута (процедурный или декларативный).

Чтобы упростить процесс спецификации реквизитов, в каждой подсистеме можно задать определенную комбинацию их значений и каждый новый атрибут подсистемы наследует эти значения. В случае необходимости значения реквизитов для отдельных атрибутов могут быть скорректированы. Помимо реквизитов каждому атрибуту ставится в соответствие пояснительная записка, содержащая определение атрибута или опорные величины для выбора его значений.

Редактор модулей и анкет позволяет модифицировать имеющиеся и создавать новые модули знаний. Особое внимание здесь уделено процедуре построения модулей. Информация эксперту предъявляется в таком виде, что каждый атрибут со своими значениями занимает ровно одну строку экрана. Если для размещения атрибута требуется более одной строки, то редактор попытается уменьшить длины имен, применяя сокращения, в крайнем случае часть значений заменяется многоточием. В процессе редактирования система обеспечивает периодическое обновление информации на экране.

Процесс построения модуля знаний сводится к манипулированию на экране строками и терминами. Редактор позволяет расставлять строки в нужном порядке и удалять те, которые не представляют интереса в данный момент. Кроме того, имеются средства для удаления отдельных значений атрибутов. Именно такой метод построения модулей способствует наиболее полному раскрытию знаний эксперта.

15.3.3. Режим консультации. Обращение к ЭС за консультацией происходит в тот момент, когда пользователь осознает необходимость получить разъяснение по какому-то вопросу. Как правило, эта проблема связывается с конкретным объектом или событием, в то время как БЗ ориентирована на определенный круг аналогичных объектов или явлений.

В общем случае консультация представляет собой последовательность сеансов, состоящих из определенного количества вопросов системы и ответов на них пользователя. В первом сеансе система выдает запрос об уточнении целей консультации. Каждый последующий сеанс консультации начинается с того, что система предлагает пользователю расширить состав экспертных групп, принимающих участие в консультации. В каждом своем вопросе система просит указать значение того или иного признака. Атрибут может использоваться в качестве вопроса, если он является обязательным или наблюдаемым нецелевым признаком или содержит по крайней мере два элемента в спектре возможных на данном этапе значений. Пользователь должен давать наиболее точные характеристики объекта. В затруднительных случаях допускаются неоднозначные ответы, в которых перечисляются целые группы значений признака.

Один из вариантов организации диалога основывается на одновременном

предъявлении всех вопросов, перечисленных в анкете. Свой ответ пользователь строит, вычеркивая ненужные значения, как это реализовано в редакторе модулей и анкет. Преимущество такого метода состоит в том, что пользователь может учесть значения многих атрибутов.

На любом шаге диалога сеанс консультации может быть прерван по желанию пользователя. Как правило, необходимость отложить диалог возникает, когда для ответа требуется информация, недоступная в настоящий момент. Следующий сеанс система начнет с вопроса, на котором произошло прерывание. Сеанс консультации может быть прерван и по инициативе системы, когда она с помощью логического вывода определяет значение процедурного атрибута.

Нормальным окончанием сеанса и всей консультации является ситуация, когда пользователь получает искомое заключение. В идеальном случае заключение оформляется в виде таблицы, в первом столбце которой указывается имя подсистемы, во втором — имя целевого атрибута и в третьем — полученное значение. Возможна ситуация, когда система придет к выводу об отсутствии целевого атрибута для данного объекта консультации. Сообщение об этом событии включается в итоговое заключение.

В ходе консультации пользователь помимо прямого ответа на поставленный вопрос может обратиться к системе за помощью или подсказкой, отказаться отвечать вообще. В ответ на запрос о помощи система выведет на экран пояснительную записку для данного атрибута. Специфика системы ФИАКР такова, что к моменту опроса по конкретному атрибуту часть его значений может быть опровергнута на основании ранее полученных данных. Однако в формулировку вопроса, чтобы придать ему объективный характер, входят все без исключения возможные значения признака. Учитывая это, пользователь может обратиться к системе за подсказкой, в которой приводится текущее состояние списка значений атрибута. Таким образом, функции помощи и подсказки в системе ФИАКР существенно различаются.

Теоретически ЭС может прервать консультацию по исчерпанию множества вопросов. Как правило, возможность такого окончания объясняется несовершенством системы знаний, а также неопределенными ответами пользователя. Заметим, что тупиковое окончание не абсолютно безнадежно: у пользователя остается шанс подключить к консультации новые экспертные группы и все-таки получить заключение системы. Переход хотя бы одного утверждения в конфликтное состояние вызывает немедленное и окончательное прерывание консультации.

Главное требование, предъявляемое к ЭС, состоит в том, что заключения системы должны быть мотивированными. Альтернативы этому тезису, вообще говоря, не существует, поскольку знания, заложенные в систему, как правило, субъективные и осознанно принять заключение можно, только получив его объяснение.

В системе ФИАКР подсистема объяснений основана на взаимодействии двух схем. Первая — схема доступа к утверждениям БЗ, которая помогает проблемному специалисту анализировать структуру БЗ (подсистема — атрибут — значение, группа — модуль и т. п.). Отметим, что такая система универсальная, т. е. не зависит от проведенной консультации. Вторая — схема трассировки — позволяет анализировать цепочку связей, возникшую в результате логического вывода. При этом подсистема запоминает пройденную пользователем трассу и позволяет возвращаться к ее отдельным фрагментам, а также отслеживать новые ветви логического вывода. В ходе трассировки система каждый раз предъявляет пользователю тот или иной модуль знаний и объясняет его функционирование в конкретной ситуации. Переходы между схемами допускаются на любом этапе объяснений. По желанию пользователя отдельные фрагменты объяснений могут выводиться на твердую печать. При формировании объяснений система использует заложенные в нее шаблоны, в которые каждый раз подставляются имена конкретных атрибутов, значений и т. д.

15.3.4. Функционирование системы ФИАКР. Содержимое БЗ может быть разбито на несколько самостоятельных текстов, каждый из которых способен

транслироваться отдельно. Транслятор с ЯПЗ системы ФИАКР является однопроходным. При обнаружении в исходном тексте ошибок загрузки модулей в БЗ автоматически отменяется. Единственное ограничение на порядок трансляции состоит в том, что модули, определяющие новые термины, должны предшествовать модулям, использующим эти термины. В процессе модификации в БЗ накапливаются видоизмененные и новые модули. Документатор, включенный в состав системы ФИАКР, позволяет выводить модифицированные модули в соответствии с соглашениями ЯПЗ.

Средства оперативной корректировки содержимого БЗ включают редактор модели проблемной области, редактор модулей и анкет, а также блок перестановок. Эксперту, вообще говоря, далеко не безразлично, в каком порядке в БЗ перечисляются подсистемы, атрибуты, модули и т. д., так как в конкретных приложениях существуют устоявшиеся схемы классификации признаков. Аналогичное замечание справедливо и по отношению к флагам. Так, порядок расположения флагов опроса непосредственно связан с процедурой управления диалогом. Блок перестановок позволяет организовать любую совокупность элементов в заданном порядке. Кроме того, этот блок осуществляет все операции уничтожения элементов БЗ. Заметим, что указанные операции весьма трудоемки. Например, при уничтожении только одного значения атрибута из БЗ необходимо исключить все модули знаний, в которые входит соответствующий дескриптор свойств.

База знаний системы ФИАКР располагается в файлах, которые имеют определенные размеры и внутреннюю структуру. Кроме того, в системе широко используются нетрадиционные средства организации диалога, связанные с особенностями реализации конкретных видеотерминалов. Физическую организацию БЗ поддерживает специальный блок обслуживания, который, в частности, выполняет роль первоотщепка при построении ЭС — создает пустые структуры БЗ и размещает их в дисковой памяти. В случае необходимости блок обслуживания осуществляет настройку БЗ на новый вид дисплеев.

Чтобы начать консультацию, пользователь должен построить отдельный экземпляр БЗ и присвоить ему имя. Система ФИАКР предлагает снабдить экземпляр помимо имени пояснительной запиской, где можно дать развернутое объяснение причин и целей данного обращения к ЭС. Если пользователь параллельно ведет несколько консультаций, текст записки позволяет ему дополнительно проконтролировать цели текущего обращения к системе. Операцию по построению экземпляра БЗ выполняет соответствующий блок из состава программного обеспечения режима проведения консультаций.

В системе ФИАКР реализована «безвозвратная» процедура логического вывода. Вместе с тем возможно возвращение к отдельным этапам диалога с целью исправления или уточнения ответов в консультационных системах такого рода необходима. В связи с этим в состав программного обеспечения режима пользователя включен дополнительный блок имитации, который позволяет развернуть весь ход прошедшей консультации и изменить некоторые ответы, обеспечивает, в частности, выход из тупиковых и конфликтных ситуаций, возникающих в ходе консультации.

По окончании консультации и получения необходимых разъяснений пользователю предоставляется возможность уничтожить экземпляр БЗ.

В режиме приобретения знаний БЗ состоит из трех файлов прямого доступа. В первом файле хранится словарь системы, во втором — размещаются утверждения. Третий файл состоит из записей разной длины, которые соответствуют либо модулям знаний, либо вспомогательным структурам. Словарь системы содержит печатные наименования всех терминов, используемых в БЗ, и пояснительные записки к выбору значений атрибутов. Во всех остальных файлах используются только логические указатели на печатные наименования. Файл утверждений представляет собой линейную последовательность утверждений, каждое из которых характеризуется своим указателем. Аналогично связи между записями в БЗ обеспечиваются адресацией их положения в файле. Каждый указатель размещается в двух байтах и интерпретируется как 16-разряд-

Таблица 15.3

Ограничения на размер модуля

Тип записи	Максимальное количество
Подсистемы	30
Атрибуты одной подсистемы	14
Значения одного атрибута	14
Анкеты	62
Экспертные группы	30
Модули одной группы	62
Целевые атрибуты	14

Таблица 15.4

Параметры ЭС ФИАКР-Т

Тип элемента БЗ	Количество
Подсистемы	28
Атрибуты подсистемы	3—12
Целевые атрибуты	9
Всего атрибутов	142
Значения атрибута	2—8
Всего дескрипторов свойств	707
Анкеты	41
Экспертные группы	12
Модули экспертной группы	15—46
Всего модулей	572

ное число. Соглашения о структуре записей позволяют интерпретировать указатели однозначно. В системе ФИАКР принят статический принцип распределения памяти. Отсюда следуют ограничения на размеры записей, а следовательно, и на размеры модулей знаний (табл. 15.3).

Ограниченность разрядной сетки, которая используется для хранения указателей, порождает ограничения на общие объемы файлов. Перевести эти ограничения в содержательные термины БЗ практически невозможно, поэтому приведем параметры ЭС ФИАКР-Т (табл. 15.4), которая использует возможный ресурс памяти на 70%.

Каждый экземпляр БЗ помимо трех основных файлов содержит четвертый, дополнительный, файл обратных ссылок на утверждения. Можно считать, что все вхождения одного и того же утверждения в экземпляре БЗ связаны цепочкой ссылок. Содержимое этого файла используется процедурой логического вывода.

Параметры генерации БЗ, ее текущие размеры, таблицы распределения памяти и другая управляющая информация содержатся в паспорте БЗ, который располагается в начале файла утверждений.

Глава 16.

Отечественные оболочки экспертных систем для больших ЭВМ

Б. С. Кирсанов, Э. В. Попов

16.1. Общее описание коммерческой гибридной системы ЭКСПЕРТИЗА

16.1.1. Назначение системы. Система ЭКСПЕРТИЗА является инструментальным средством (ИС), предназначенным для создания экспертных систем (ЭС), решающих статические неформализованные задачи в различных проблемных областях. ЭКСПЕРТИЗА может быть охарактеризована как коммерческое гибридное средство типа настраиваемой оболочки ЭС (см. гл. 13). (Полное описание системы представлено в документации на изделие [Изделие ЭКСПЕРТИЗА].)

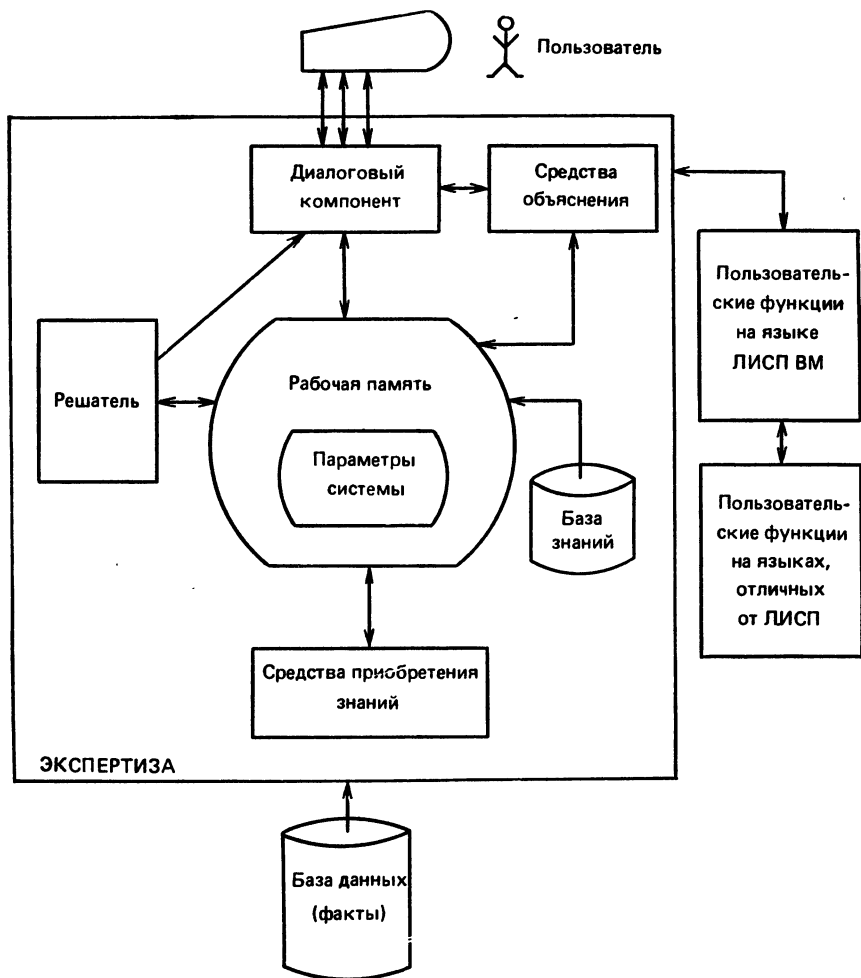


Рис. 16.1. Структура системы ЭКСПЕРТИЗА

Создание конкретной ЭС (приложения) средствами системы ЭКСПЕРТИЗА достигается вводом знаний о проблемной области в ее базу знаний (БЗ) и выбором параметров, обеспечивающих настройку системы на особенности конкретной проблемной области. Создание приложения не требует от эксперта, вводящего знания о проблемной области, умения программировать.

Приложения, созданные с помощью системы, могут включать любые пользовательские функции, написанные на языке программирования ЛИСП ВМ, на котором реализована система ЭКСПЕРТИЗА. При этом допускается вызов из функций, написанных на языке ЛИСП, и функций (подпрограмм), реализованных на других языках программирования.

16.1.2. Структура системы. Система ЭКСПЕРТИЗА осуществляет решение задач на основе согласованного использования информации от пользователя,

из базы знаний, из базы данных (типа ADABAS) и от пользовательских функций (рис. 16.1). Взаимодействие системы ЭКСПЕРТИЗА с базой данных (БД) и пользовательскими функциями позволяет охарактеризовать ее как гибридное инструментальное средство ЭС, объединяющее технологию инженерии знаний, БД и прикладных программ.

Функционально система включает все основные компоненты традиционной ЭС (см. гл. 13): диалоговый компонент, обеспечивающий взаимодействие с пользователем; компонент приобретения знаний, обеспечивающий ввод знаний в ЭКСПЕРТИЗУ при создании конкретной ЭС; решатель, осуществляющий на основе имеющихся знаний решение задач; компонент объяснения, обеспечивающий отладку знаний и доверие пользователя к полученным результатам; БЗ, содержащую знания о решении задач (правила) и факты (значения атрибутов объектов) о конкретных проблемных ситуациях; рабочую память (РП), содержащую данные о текущей проблемной ситуации в виде значений атрибутов объектов, которые служат для представления характеристик сущностей реального мира.

В отличие от традиционной схемы в РП системы ЭКСПЕРТИЗА хранятся также параметры системы, интерпретируемые ею в ходе работы. Изменяя их значения, можно управлять функционированием решателя. Система ЭКСПЕРТИЗА допускает использование при решении задач внешних (пользовательских) функций или программ, реализованных на языке ЛИСП или любом другом языке, и данных из БД.

Система ЭКСПЕРТИЗА функционирует в двух режимах: режиме приобретения знаний и режиме консультации, в ходе которого пользователь, эксперт или инженер по знаниям взаимодействует с ЭС для решения задачи. При создании ЭС используются поочередно оба режима, а при использовании — только режим консультации.

16.2. Возможности режима приобретения знаний

16.2.1. Формирование базы знаний. Основной задачей режима приобретения знаний является формирование БЗ конкретной ЭС. Для этого в БЗ вводятся следующие элементы: описание объектов; правила, метаправила; описание предикатных функций; описание вычислительных функций; факты.

Эксперт и инженер по знаниям определяют типы понятий проблемной области, использование которых необходимо (полезно) для решения задач. В системе ЭКСПЕРТИЗА принята модель проблемной области, в которой состояние может описываться переменным числом одно- и разнотипных понятий. Каждое состояние проблемной области представляется конкретным набором понятий с конкретными значениями характеристик. Для представления понятий и характеристик проблемной области в системе используются объекты и их атрибуты.

Описание объектов задает:

- тип объекта для обозначения типа понятия, которое он представляет;
- описание значений атрибутов, которые определяют возможные конкретные объекты, представляющие конкретные понятия;
- тип атрибутов, определяющий способ их использования при решении задач;
- ограничения на значения атрибутов для контроля полученных значений;
- значения по умолчанию, используемые для предположений о значениях атрибутов, когда системе значения атрибутов неизвестны;
- лингвистическую информацию, используемую системой при анализе (синтезе) сообщений пользователя.

Для представления состояния системы и значений ее параметров существует стандартный управляющий объект.

На основе описаний объектов в РП при решении задач создаются конкретные объекты с конкретными значениями атрибутов. Кроме значений атрибутов

конкретных объектов в РП имеются их свойства, вырабатываемые и используемые решателем.

Для представления зависимостей между значениями атрибутов объектов используются правила, состоящие из левой и правой частей. Левая часть включает предпосылку и условие. Предпосылка описывает абстрактную ситуацию проблемной области, наличие которой необходимо для рассмотрения правила. Условие указывает, соблюдение каких соотношений в выделенной предпосылочной ситуации достаточно для выполнения правой части правила. Предпосылка задается совокупностью объектов и взаимосвязями между их атрибутами. Условие задается в виде конъюнкции предикатных функций, определенных на значениях атрибутов объектов. Применение правила состоит в следующем. Конкретные объекты, существующие в текущий момент в РП, сопоставляются с левой частью правила. Этот процесс проходит две стадии. Сначала определяется наличие в РП ситуации, соответствующей предпосылке правила; если ситуация существует, то проверяется, выполняется ли условие правила на подставленных из выделенной ситуации значениях. Если подстановка (означивание), удовлетворяющая условию, найдена (условие выполнено), то это означает возможность вычисления действия (вычислительной функции), определенного в правой части правила. Действие правой части выполняется при значениях атрибутов, взятых из левой части правила.

Правила разрешается разбивать на классы, соответствующие подзадачам задачи или разным методам ее решения. В каждый текущий момент времени решатель работает с одним классом правил.

Правила и значения атрибутов (факты) могут снабжаться коэффициентом определенности, характеризующим доверие к ним, измеряемое числом (десятичной дробью) в диапазоне от 0 до +1.

Метаправила используются для представления знаний о ходе решения задачи, позволяя эксперту и инженеру по знаниям разбивать исходную задачу на подзадачи, решать их независимо и планировать последовательность решения подзадач. Областью действий метаправил являются системные процессы. Структура метаправил аналогична структуре правил, только в них дополнительно разрешается использовать предикатные функции, определенные на системных свойствах атрибутов и управляющем объекте, и функции (в правой части правила), имеющие область действия знания и системные процессы. Набор предикатных и вычислительных функций в метаправилах фиксирован.

Предикатные функции (в левых частях правил) используются с целью проверки выполнимости условия правила. Выполнимость условия оценивается коэффициентом определенности, вычисляемым в соответствии с определенностью значений предикатных функций и их аргументов по формулам системы MYCIN [Shortliffe, 1976]. Условие считается выполненным, если полученный коэффициент определенности положительный (или больше установленного порога). Вычислительной функцией может быть любая функция языка ЛИСП, в том числе и вызывающая программы, реализованные на других языках программирования.

Факты несут в себе информацию о состояниях проблемной области, с помощью которой приписываются значения атрибутам объектов. С помощью фактов задается обобщающая (справедливая для всех или для значительного числа конкретных состояний проблемной области) информация. Если количество фактов, необходимых для задания состояний предметной области, не превышает 1000 записей конкретных объектов, то их хранение на внешнем носителе в виде списков языка ЛИСП обеспечивается системой. В противном случае факты должны храниться в БД. В виде списков факты хранятся аналогично записям БД реляционного типа. Каждую запись (список) можно рассматривать как реализацию объекта. Структура записи БД должна быть представлена как описание объекта. Система ЭКСПЕРТИЗА обеспечивает генерацию запросов на поиск и выдачу фактов из БД на языке системы управления БД типа ADABAS. Совокупность фактов о проблемной области, собранных в РП в виде значений атрибутов объектов и полученных от пользователя из БД и в результате работы системы, составляют данные о решаемой задаче (проблемной ситуации).

Средства приобретения знаний обеспечивают ввод, каталогизацию, хранение, просмотр и корректировку рассмотренных элементов БЗ. В ходе приобретения знаний, чтобы обеспечить их сопряженность, проводится семантико-синтаксический контроль.

16.2.2. Выбор параметров системы. Дополнительной задачей режима приобретения знаний является выбор значений параметров системы, осуществляемый с целью повышения эффективности решения задач без изменения БЗ. Для некоторых проблемных областей удачный выбор значений параметров позволяет в 2—3 раза сократить время работы. (Используемые в системе параметры и их назначение излагаются при обсуждении возможностей системы по решению задач в § 16.3). Оптимальный выбор значений параметров требует тщательного анализа динамических, статических и статистических характеристик проблемной области. Однако в настоящее время методика по выбору значений параметров отсутствует. Поэтому предлагается после тестирования ЭС подбирать значения параметров экспериментально. С этой целью реализована возможность предоставления пользователю некоторой статистики и временной трассы решения задач.

Система ЭКСПЕРТИЗА может с помощью метаправил осуществлять динамическое (т. е. в ходе решения задачи) изменение значений параметров решателя.

16.2.3. Язык представления знаний (ЯПЗ). Предназначен для записи знаний эксперта о проблемной области. Позволяет представлять знания как декларативно — в виде описаний объектов и фактов, так и процедурно — в виде правил, метаправил и функций. При описании ЯПЗ используется расширенная бэкусовская нотация: в квадратных скобках приводятся необязательные элементы, в фигурных — альтернативные, разделяемые символом (или); круглые скобки и любые символы не в угловых скобках обозначают сами себя. Многоточием обозначается возможный многократный повтор предшествующего элемента; в угловых скобках указываются метaperменные. Все символы, состоящие из прописных букв, обозначают элементы ЯПЗ.

Описание начинается с декларативной составляющей языка, определяющей используемые объекты:

```
<описание объектов> ::= ((<описание объекта>) [(<описание объекта>) ...])
```

```
<описание объекта> ::= (<имя объекта>) [(<комментарий>)]  
[(ТИП {ПРЕДИКАТ | ПОНЯТИЕ})] (<описание атрибута>) ...
```

Имя объекта должно быть уникально; тип объекта характеризует тип описываемого объектом понятия, используется только при синтезе сообщений.

```
<комментарий> ::= <строка>
```

```
<имя объекта> ::= <идентификатор>
```

```
<строка> — строка символов длиной не более 256 символов.
```

Описание атрибута объекта служит для определения некоторой характеристики понятия, представляемой данным объектом:

```
<описание атрибута> ::= (<имя атрибута>) [(<комментарий>)]  
(<описание значений>) [(<свойства атрибута>)]
```

```
<имя атрибута> ::= <идентификатор>
```

Имя атрибута должно быть уникально в пределах объекта.

```
<описание значений> ::= ТИП {СИМВОЛ <диапазон символьный> |  
ЧИСЛО <диапазон числовой> | ДАТА <диапазон даты> | <объект. атрибут> | ЛЮБОЙ} [ПО_УМОЛЧАНИЮ <список значений>]
```

```
<диапазон символьный> ::= {ЛЮБОЙ | ((<строка>) ...)}
```

```
<диапазон числовой> ::= (ОТ <число> ДО <число>) ...
```

```
<число> — любое число, содержащее не более 16 десятичных цифр с точкой
```

```
<диапазон даты> ::= (ОТ <дата> ДО <дата>) ...
```

```
<дата> ::= <ЧЧ> <ММ> <ГГ>
```

$\langle \text{ЧЧ} \rangle$ — число от 1 до 31
 $\langle \text{ММ} \rangle$ — число от 1 до 12
 $\langle \text{ГГ} \rangle$ — число от 1 до 99
 $\langle \text{объект. атрибут} \rangle ::= (\langle \text{имя объекта} \rangle . \langle \text{имя атрибута} \rangle)$

Слово ЛЮБОЙ означает, что ограничений на значения символического атрибута не задается, в противном случае перечисляются все возможные значения, каждое из которых может быть длиной строкой. Числовые атрибуты и даты определяются совокупностью интервалов, характеризующих диапазон их возможных значений.

Значение можно задать ссылкой на ранее определенный атрибут того же самого или другого объекта, т. е. описание значения будет заимствовано из того атрибута, на который сделана ссылка.

Характеристика ПО_УМОЛЧАНИЮ задает одно или несколько значений, которые будут использоваться как предположения о значении атрибута в ситуациях, когда система не может вывести значения искомого атрибута. Если у атрибута указано несколько значений, то будет создано несколько альтернативных миров.

$\langle \text{свойства атрибута} \rangle ::= [\text{ИТЕРАТИВНОСТЬ} \{ \text{ДА} \mid \text{НЕТ} \}],$
 $[\text{ОБЯЗАТЕЛЬНОСТЬ} \{ \text{ОБ} \mid \text{НО} \}], [\text{НАЧАЛЬНОЕ\_ЗНАЧЕНИЕ}]$
 $\{ \text{СПР} \mid \text{НЕСПР} \}, [\text{ЕЯ\_ЗАПРОС\_О\_ЗНАЧЕНИИ} \langle \text{строка} \rangle],$
 $[\text{ЕЯ\_ОТВЕТ\_О\_ЗНАЧЕНИИ} \langle \text{строка} \rangle]$

Здесь свойство ИТЕРАТИВНОСТЬ означает, что возможно итеративное вычисление значения данного атрибута (через то же значение). Значение такого атрибута считается определенным только после того, как для его вычисления применялись все возможные правила. Только такие атрибуты разрешается использовать в условиях правил вычисления их же значений. Значение ОБ свойства ОБЯЗАТЕЛЬНОСТЬ означает способность атрибута однозначно идентифицировать объект своим значением, а значение НО указывает, что атрибут однозначно объект не определяет. Свойство НАЧАЛЬНОЕ ЗНАЧЕНИЕ определяет, следует ли при начальном вводе спрашивать (СПР, НЕСПР) пользователя о значении атрибута. Два оставшихся свойства задают шаблоны, определяющие, как должен быть представлен на русском языке вопрос к пользователю о значении атрибута и как значение атрибута должно представляться пользователю. Шаблон задается символической строкой, внутри которой вместо значения атрибута разрешается использовать символы $\langle 3\_A \rangle$. При синтезе ответа вместо $\langle 3\_A \rangle$ будет подставлено вычисленное значение атрибута. Приведем пример описания объекта:

((Пациент) (ТИП ПОНЯТИЕ)
 (ФИО (Фамилия-имя-отчество)
 (ТИП СИМВОЛ ЛЮБОЙ)
 ОБЯЗАТЕЛЬНОСТЬ ОБ
 ЕЯ ЗАПРОС О ЗНАЧЕНИИ (Каковы фамилия, имя, отчество пациента)
 ЕЯ ОТВЕТ О ЗНАЧЕНИИ (Фино пациента $\langle 3\_A \rangle$))
 (ПОЛ
 (ТИП СИМВОЛ ((М) (Ж)))
 (ГОД РОЖДЕНИЯ
 (ТИП ЧИСЛО ОТ 1880 ДО 1990)
 ЕЯ ЗАПРОС О ЗНАЧЕНИИ (Каков год рождения?),
 ЕЯ ОТВЕТ О ЗНАЧЕНИИ (Год рождения $\langle 3\_A \rangle$))
 ...))

Описания объектов определяют вид сущностей (их характеристики и значения), которые могут иметь место в проблемной области. То, какие конкретные объекты (значения их атрибутов) имеют место в проблемной области, задается в виде базы фактов.

База фактов может быть реализована как БД типа ADABAS либо как список фактов (конкретных объектов), поддерживаемых системой:

<список фактов> ::= (<отношение>) ...
<отношение> ::= (<имя отношения>) (<запись>) ...
<запись> ::= (<имя атрибута> . <значение атрибута>) ...
<имя отношения> ::= <идентификатор>
<имя атрибута> ::= <идентификатор>
<значение атрибута> ::= {<число> | <строка>}

Приведем пример базы фактов в виде списков ЛИСП:

(Сотрудник ((ФИО. ПЕТРОВ)	(Д. РОЖДЕНИЯ. 1953)
(ПОЛ. М))	
((ФИО. СИДОРОВ)	(Д. РОЖДЕНИЯ. 1960)
(ПОЛ. М)) ...)	
(Заболевание ((КТО. ИВАНОВ)	(КОГДА. 1983)
(ТЕЧЕНИЕ ТЯЖЕЛОЕ)	(ТИП. ИНФЕКЦИОННОЕ)
...)	

Перейдем к описанию процедурной части ЯПЗ:

<правило> ::= (<левая часть>) (<правая часть>)
<левая часть> ::= (<имя правила>) (<приоритет>) [ПОЛН] [СОСТ]
<определенность> ((<предпосылка>) (<условие>))
<имя правила> ::= <идентификатор>
<приоритет> — число от 0 до 9999

Признак ПОЛН означает, что правило принимает участие в получении решения только в случае полного означивания, признак СОСТ означает, что правило предназначено для определения состава проблемной области. В этом случае результат работы правила приводит к появлению в РП нового объекта, как в случаях функции ИСКАТЬ1 и СПРОСИ (см. п. 16.3.1).

<определенность> — десятичная дробь, принимающая значение в диапазоне от 0 до 1;

<предпосылка> ::= (<элемент предпосылки>) [(<элемент предпосылки>) ...]

<элемент предпосылки> ::= <имя объекта> ((<имя атрибута>, <переменная>) ...)

<переменная> — строка длиной до трех символов.

Предпосылка (см. п. 16.2.1) задается перечислением имен объектов (типов объектов) и атрибутов, значения которых должны присутствовать в РП. Переменные являются ссылками на значение атрибутов и указывают взаимосвязи между объектами и их атрибутами. Если значения каких-либо двух атрибутов разных объектов обозначены одинаковыми переменными, то это означает, что объекты по этим атрибутам связаны, т. е. при вычислении левой части эти атрибуты должны иметь одинаковые значения. Кроме того, переменные используются для ссылки на значения атрибутов в шаблонах вызова предикатных и вычислительных функций.

<условие> ::= { (Т) | <шаблон вызова предикатной функции> } [(<шаблон вызова предикатной функции>) ...].

Здесь условие состоит из совокупности вызовов предикатных функций, оформленных как шаблон вызова, в котором в качестве аргументов стоят переменные. Считается, что все предикатные функции соединены связкой &; (Т) означает, что условие всегда выполняется.

Опишем предикатные функции, которые могут использоваться в правилах:

<предикатная функция> ::= <тело функции> [(<комментарий>)]
<шаблон вызова предикатной функции>

<тело функции> — это текст программы на языке ЛИСП;

<шаблон вызова предикатной функции> ::= {<имя предикатной функции> (<аргумент>) | <имя предикатной функции> ((<аргумент>) (<аргумент>)) }

Шаблоны вызова предикатных функций

№ п/п	Шаблон вызова	Комментарий
1	РАВНО ((A1) (A1))	Аргументы (A1 и A2) любого типа
2	НЕ РАВНО ((A1) (A1))	То же
3	БОЛЬШЕ ((A1) (A2))	A1 и A2 могут быть числом и датой
4	МЕНЬШЕ ((A1) (A2))	То же
5	БОЛЬШЕ = ((A1) (A2))	"
6	МЕНЬШЕ = ((A1) (A2))	"
7	ВХОДИТ ((A1) (A2))	A2 — список; функция равна 1, если A1 входит в список A2
8	НЕ ВХОДИТ ((A1) (A2))	Функция, обратная функции № 7
9	ОДИН ИЗ ((A1) (A2))	Функция, эквивалентная функции № 7

$\langle \text{имя предикатной функции} \rangle ::= \langle \text{идентификатор} \rangle$

$\langle \text{аргумент} \rangle ::= \langle \text{идентификатор} \rangle$

Шаблоны вызова указывают вид функции, которая вызывается из правила. В табл. 16.1 приведены шаблоны вызова предикатных функций и комментарии к ним.

Правая часть правила определяет способ получения значений атрибутов: с помощью вычислений; из базы данных (ИСКАТЬ); от пользователя (ВОПРОС):

$\langle \text{правая часть} \rangle ::= (\langle \text{цель} \rangle) (\{ \langle \text{шаблон вызова вычислительной функции} \rangle | \text{ИСКАТЬ} \{1|2\} \langle \text{условия поиска} \rangle | \text{ВОПРОС} \{1\ 2\} \langle \text{контекст вопроса} \rangle \})$

$\langle \text{цель} \rangle ::= \langle \text{переменная} \rangle$

Семантика функций ИСКАТЬ и ВОПРОС рассмотрена в п. 16.3.1.

$\langle \text{вычислительная функция} \rangle ::= \langle \text{тело функции} \rangle [(\langle \text{комментарий} \rangle)]$
 $\langle \text{шаблон вызова вычислительной функции} \rangle$

$\langle \text{шаблон вызова вычислительной функции} \rangle ::= \langle \text{имя вычислительной функции} \rangle [(\langle \langle \text{аргумент} \rangle \dots \rangle)]$

$\langle \text{имя вычислительной функции} \rangle ::= \langle \text{идентификатор} \rangle$

Шаблоны вызова вычислительных функций, кроме ИСКАТЬ, ВОПРОС и функций метаправил, приведены в табл. 16.2.

В правой части правила может быть и вычислительная, и предикатная функция. Использование вычислительных функций в левой части правила не разрешается, так как приведет к ее многократному вычислению в ходе сопоставления даже тогда, когда это не требуется для получения результата, без этого можно обойтись, введя атрибут, значением которого является значение функции, которую хотелось бы использовать в левой части правила.

$\langle \text{условие поиска} \rangle ::= (\langle \text{искомое} \rangle) (\langle \text{условие} \rangle)$

Условие поиска задает, что надо выбирать (какой факт из БД), и условия, по которым факт надо отбирать.

$\langle \text{искомое} \rangle ::= \{ \langle \text{имя функции} \rangle . \langle \text{аргумент} \rangle | \langle \text{имя отношения} \rangle . \langle \text{атрибут} \rangle \}$

Шаблоны вызова стандартных вычислительных функций

Шаблон вызова	Комментарий
К	К — константа, значением функции является К
ПРИСВОИТЬ (A1)	A1 — переменная, значением функции является значение переменной A1
СЛОЖИТЬ (A1) ... (AN)	Число аргументов переменное, результатом является сумма всех аргументов, аргументы имеют тип ЧИСЛО
ВЫЧЕСТЬ ((A1) (A2))	В качестве второго аргумента может быть функция, тип аргументов — ЧИСЛО
УМНОЖИТЬ ((A1) (A2))	То же
ДЕЛИТЬ ((A1) (A2))	"
ПРОЦЕНТ ((A1) (A2))	"
ВЫЧЕСТЬ ДАТЫ ((A1) (A2))	A1 и A2 имеют тип ДАТА, значение функции — ЧИСЛО
ДОБАВИТЬ К ДАТЕ ((A1) (A2))	A1 — тип ДАТА, A2 — тип ЧИСЛО; значение функции — ДАТА
ВЫЧЕСТЬ ЧИСЛО ((A1) (A2))	A1 — тип ДАТА, A2 — тип ЧИСЛО, значение функции — ДАТА

Первая альтернатива используется, когда факты хранятся во внешней СУБД типа ADABAS. Имя функции указывает вычислительную функцию, которая реализует сопряжение системы с СУБД. В качестве аргументов ей передается имя атрибута, подлежащего выборке, и условие, которое рассматривается как условие поиска. Вторая альтернатива указывает, из какого отношения и какой атрибут нужно выбирать.

$\langle \text{контекст вопроса} \rangle ::= (\langle \text{переменная} \rangle) [(\langle \text{переменная} \rangle) \dots]$

Первая переменная указывает тему вопроса, последующие определяют вид, в котором запрос об этой теме задается. Если последующие переменные не указаны, то контекст вопроса формируется системой в соответствии с описанием объектов.

Особенностью метаправил (м.п.) по сравнению с правилами является возможность расширения левой части правила системными условиями и специфический набор действий правой части:

$\langle \text{метаправило} \rangle ::= (\langle \text{левая часть м.п.} \rangle \langle \text{правая часть м.п.} \rangle)$

$\langle \text{левая часть м.п.} \rangle ::= (\langle \text{имя правила} \rangle) (\langle \text{приоритет} \rangle \text{МЕТА})$
 $((\langle \text{предпосылка} \rangle) (\langle \text{условие проблемное} \rangle) (\langle \text{условие системное} \rangle))$

$\langle \text{условие проблемное} \rangle ::= \langle \text{условие} \rangle$

$\langle \text{условие системное} \rangle ::= \{(T) | \langle \text{условие по свойствам} \rangle\} \{(T) | \langle \text{условие по целям} \rangle\}$

$\langle \text{условие по свойствам} \rangle ::= (\langle \text{проверка свойства} \rangle) [(\langle \text{проверка свойства} \rangle) \dots]$

$\langle \text{проверка свойства} \rangle ::= \{ \text{НЕТ} - \text{ВОПРОС} \langle \text{переменная} \rangle \} | \text{НЕТ} - \text{ИСКАТЬ} \langle \text{переменная} \rangle | \text{ТУПИК} \langle \text{переменная} \rangle \}$.
 Свойства вырабатываются интерпретатором и приписываются атрибутам конкретных объектов, расположенных в РП. $\text{НЕТ\_ВОПРОС}(X)$, $\text{НЕТ\_ИСКАТЬ}(X)$, $\text{ТУПИК}(X)$ — это предикаты, принимающие значение «истина», если у атрибута X , на который ссылаются с помощью переменной, присутствует соответствующее свойство. Считается, что проверки свойств связаны конъюнкцией. Предикат $\text{НЕТ\_ВОПРОС}(X)$ имеет значение «истина», если пользователь не ответил на вопрос о значении атрибута X , а предикат НЕТ_ИСКАТЬ , если значение не было найдено в БД. Предикат $\text{ТУПИК}(X)$ имеет значение «истина», если система с помощью имеющихся правил не смогла вычислить значение атрибута X .

$\langle \text{условие по целям} \rangle ::= (\langle \text{целевое условие} \rangle) [(\langle \text{целевое условие} \rangle) \dots]$

$\langle \text{целевое условие} \rangle ::= \{ \langle \text{имя объекта} \rangle | \langle \text{имя объекта} \rangle . \langle \text{имя атрибута} \rangle \}$

Условие по целям имеет значение «истина», если система в ходе достижения цели выставила подцели в порядке, указанном в метаправиле. Целевое условие указывает, при достижении каких целей уместно применять данное метаправило. Обычно целевое условие используется, чтобы проверить, в рамках какой задачи вызывается подзадача. Имя объекта указывает, что если текущей или предшествующей целью было вычисление значения атрибута объекта указанного типа, то условие по целям выполнится. Если указан еще и атрибут, то целью должно быть вычисление значения указанного атрибута.

Правая часть метаправила предназначена для задания функций, определяющих: знания, необходимые системе для решения задачи в целом; правила, которые следует использовать в текущий момент работы системы; порядок использования правил при решении задачи:

$\langle \text{правая часть м.п.} \rangle ::= \{ \langle \text{функция загрузки} \rangle | \langle \text{функция перехода} \rangle | \langle \text{функция управления} \rangle \}$

$\langle \text{функция загрузки} \rangle ::= \text{ЗАГРУЗИТЬ} \langle \text{описание классов} \rangle \langle \text{разделы функций} \rangle \langle \text{разделы описания объектов} \rangle \langle \text{разделы базы данных} \rangle$

$\langle \text{описание классов} \rangle ::= (\langle \text{описание класса} \rangle) [(\langle \text{описание класса} \rangle) \dots]$

$\langle \text{описание класса} \rangle ::= \langle \text{имя класса} \rangle ((\langle \text{раздел правил} \rangle) [(\langle \text{раздел правил} \rangle) \dots])$

$\langle \text{имя класса} \rangle ::= \langle \text{идентификатор} \rangle$

Функция загрузки определяет знания, с которыми будет работать система до применения следующего аналогичного метаправила, а также имена классов правил и их состав с указанием совокупности разделов, в которых правила указанного класса хранятся в базе знаний, и подмножества функций, описаний объектов и отношений базы данных, имеющих отношение к данной задаче. Ссылка на все элементы БЗ осуществляется указанием имен разделов библиотеки, в которых хранятся знания. Имена разделов задаются символьной строкой длиной до семи символов. Метаправила с функцией загрузки служат для выбора и определения решаемой задачи, включаются в набор правил (класс), который загружается в начале работы системы.

$\langle \text{функция перехода} \rangle ::= \{ \text{ИДТИ} [(C)] \langle \text{имя класса} \rangle | \text{ВЫЗВАТЬ} [(C)] \langle \text{имя класса} \rangle \text{ВЕРНУТЬСЯ} [(C)] \}$

Функция ИДТИ Y , где Y — имя класса правил, вызывает переход с текущего класса правил X к работе с указанным классом правил Y . Необязательный признак (C) означает, что при выполнении функции надо удалить свойство ТУПИК у всех атрибутов, у которых оно появилось в текущем классе. Признак (C) используется в тех случаях, когда одно и то же значение последовательно

пытаются вывести различными методами (классами правил). Дело в том, что решатель не вычисляет значения атрибутов (не выставляет их как подцели), имеющих свойство ТУПИК. Функция перехода ВЫЗВАТЬ У аналогична функции ИДТИ, но обеспечивает возможность возврата по функции ВЕРНУТЬСЯ в текущий класс X.

$\langle \text{функция управления} \rangle ::= \{ \text{ИСКЛЮЧИТЬ}(\langle \text{список правил} \rangle) | \text{ПЕРВЫМИ}(\langle \text{список правил} \rangle) | \text{ПОСЛЕДНИМИ}(\langle \text{список правил} \rangle) | \text{РАНЫШЕ}((\langle \text{список правил} \rangle)(\text{список правил})) | \text{СОЗДАТЬ} \langle \text{аргументы создать} \rangle) | \text{ПРОТИВОРЕЧИЕ} \}$

$\langle \text{список правил} \rangle ::= (\langle \text{имя правила} \rangle) [(\langle \text{имя правила} \rangle) \dots]$

Функции ИСКЛЮЧИТЬ, ПЕРВЫМИ, ПОСЛЕДНИМИ, РАНЫШЕ изменяют порядок выполнения (приоритеты правил) в данном классе на время активности этого класса правил. В результате выполнения некоторой функции управления приоритет всех правил, указанных в списке правил, изменяется по отношению ко всем другим правилам активного класса. При этом приоритеты правил, указанных в метаправиле, по отношению друг к другу не изменяются. Функция СОЗДАТЬ образует в рабочей памяти объект с указанным типом и значениями атрибутов:

$\langle \text{аргументы создать} \rangle ::= (\langle \text{имя объекта} \rangle) (\langle \text{имя атрибута} \rangle . \langle \text{переменная} \rangle) [(\langle \text{имя атрибута} \rangle . \langle \text{переменная} \rangle) \dots]$

Функция ПРОТИВОРЕЧИЕ используется для прекращения работы с альтернативными мирами при обнаружении их противоречивости, которая явилась следствием сделанных ранее предположений. Факт противоречивости фиксируется выполнением условия метаправила. Функция может также использоваться для проверки непротиворечивости БЗ и ответов пользователя.

Приведем примеры правил и метаправил:

(П1	(100 0.8)	((ПАЦИЕНТ	(ФИО Ч) (ЗДОРОВЬЕ ?)) (ЗАБОЛЕВАНИЕ (КТО — БОЛЕЛ Ч) (ТИП Т) (КОГДА Г1)) (ЗАБОЛЕВАНИЕ (КТО — БОЛЕЛ Ч) (КОГДА Г2))) ((РАВНО (Г1) (Г2)) (РАВНО (Т) (ТЯЖЕЛОЕ)))
	((?) ПЛОХОЕ))		
(П2	(101 МЕТА)	(ПАЦИЕНТ	(ФИО Ч) (ПОЛ М)) (Т) (Т) (ПАЦИЕНТ. ИЗБЫТОЧНЫЙ—ВЕС) (ИЗБЫТОЧНЫЙ—ВЕС))
	(ВЫЗВАТЬ		
П3	(101 МЕТА)	(ПАЦИЕНТ	(ФИО Ч) (ПОЛ П) (ГОСТ Р)) (Т) (ТУПИК (Р))
	(ВЕРНУТЬСЯ))		

16.3. Возможности режима консультации

16.3.1. Назначение режима консультации. В режиме консультации осуществляется кооперативное решение задачи пользователем и ЭС. Первоначально инициатива в диалоговом взаимодействии принадлежит ЭС, которая осущест-

Структура метадialoga

Имя диалога	Комментарий
ВЫБОР ЗАДАЧИ ПОСТАНОВКА ЗАДАЧИ КОНСУЛЬТАЦИЯ КОРРЕКЦИЯ	Выбирается задача для решения Вводятся исходные данные задачи Выполняется решение задачи Идет диалог по устранению ошибки в ответе пользователя
ПРЕДОСТАВЛЕНИЕ ОТВЕТА	Пользователю сообщаются результаты решения задачи
ПРОСМОТР РАБОЧЕЙ ПАМЯТИ ВВОД	Просматриваются результаты работы системы Осуществляется дополнительный ввод данных в рабочую память
ЧТО ЕСЛИ	Корректируется состояние рабочей памяти с целью повторного решения задачи при измененных исходных или промежуточных данных
ОБЪЯСНЕНИЕ	Пользователю объясняется процесс функционирования системы и ее знания
ПЕРЕХОД	Осуществляется выбор для анализа одного из построенных ЭС альтернативных миров

влет разбienie задачи на подзадачи. Часть подзадач решает ЭС, а часть — пользователь. Система обращается к пользователю в тех случаях, когда она или не может решить подзадачу, или ей известно, что пользователь может это сделать лучше. Подзадача рассматривается как вычисление значения некоторого атрибута.

Кооперативное решение задачи в режиме консультации обеспечивается благодаря: поддержанию взаимодействия с пользователем в соответствии с заранее определенной структурой метадialoga по ведению консультации; получению информации о проблемной области от пользователя, от внешних программ и из базы данных; решению задач в условиях неопределенности исходных данных и знаний, которое достигается нахождением и реализацией алгоритма решения задачи в контексте конкретных данных о проблемной области, при этом находятся все возможные решения задачи, которые снабжаются коэффициентом определенности, оценивающим решение; решению задач в условиях неполноты данных и знаний, которое достигается выставлением предположений и удалением возникающих противоречий. Структура метадialoga приведена в табл. 16.3.

Диалог о проблемной области состоит из взаимосвязанных высказываний участников, имеющих целью ввести в систему информацию о проблемной области (решение подзадач). Структура проблемного диалога (порядок и состав вопросов о значении атрибутов объектов) задается содержимым БЗ (в виде правил) и различна в каждой конкретной ЭС. В нее входят вопрос-ответные сообщения в форме анкет, меню ЕЯ-текстов. Форма выдаваемых сообщений определяется системой автоматически. Инициатива в диалоге принадлежит системе. Перехватить инициативу пользователь может, выдавая специальные команды вместо ответов на сообщения системы.

Ведение консультации осуществляется в соответствии с жестко встроенной структурой метадialoga, задаваемой в виде ориентированного графа. Вершинам графа соответствуют подdialogи (один из которых — диалог о проблемной области по решению задач), а дуги указывают возможные переходы от

диалога к диалогу, которые осуществляются на основе команд, выдаваемых пользователем. Каждая вершина графа, задающего структуру метadiaлога, может разворачиваться в подграф, определяющий поддиалоги и порядок их смежности в рамках диалога. В рамках каждого диалога и поддиалога система сообщает пользователю список возможных команд.

Все диалоги, кроме проблемного, который инициируется автоматически, начинаются по командам пользователя. Завершение текущего диалога приводит к возврату в диалог предшествующий. Возврат к диалогу о проблемной области может осуществляться из любого поддиалога непосредственно.

Получение исходной информации происходит в начале диалога о проблемной области, когда на экране видеотерминала представляется объект, интересующий пользователя. При формулировании задачи пользователь указывает значения исходных атрибутов и знаком вопроса помечает атрибуты, значения которых необходимо вычислить. Затем система начинает решение задачи. Получение дополнительной информации о проблемной области осуществляется с помощью правил, имеющих в правой части специальную функцию ВОПРОС или ИСКАТЬ. Левая часть правила должна описывать ситуации, в которой целесообразно задавать вопрос или осуществить поиск фактов. «Целесообразно» в том смысле, чтобы был сформирован контекст, обеспечивающий достаточно однозначный и не лишенный смысла ответ на вопрос или результат поиска в БД.

Возможно получение информации о проблемной ситуации двух видов. Это могут быть вопросы, во-первых, о понятиях проблемной области и, во-вторых, о значениях характеристик понятий. Тип вопроса определяет эксперт, формирующий БЗ. В первом случае используются функции ВОПРОС1 и ИСКАТЬ1, во втором ВОПРОС2 и ИСКАТЬ2. Различие в вопросах проявляется, если пользователь сообщает неоднозначный ответ. При ВОПРОС1 ЭС, получив более одного ответа, создает в описании проблемной ситуации несколько объектов, представляющих сущности. При ВОПРОС2 ЭС, получив более одного ответа, каждый из которых неточный, создает несколько альтернативных пространств, в которых один и тот же атрибут имеет различные значения. Аналогичные различия имеются между функциями ИСКАТЬ1 и ИСКАТЬ2.

Напомним, что значения атрибутов в общем случае могут быть получены от пользователя; извлечены из БД; вычислены с помощью правил. Порядок определения значений может устанавливаться экспертом с помощью расстановки приоритетов у соответствующих правил. Выбор порядка вычисления значений атрибутов определяет порядок ведения диалога о проблемной области.

В системе реализован режим работы (его использование устанавливается значением параметра), при котором для всех атрибутов, значения которых система вычислить не может, автоматически генерируется вопрос к пользователю (всегда ВОПРОС2). Этот режим полезен при отладке системы и позволяет строить ЭС без получения от эксперта информации о разделении ролей между пользователем и ЭС.

В ходе решения задачи с целью минимизации вопросов система задает пользователю только общие (а не конкретные) вопросы о значении атрибута объекта и способна принимать от пользователя только прямые ответы на вопросы. В ходе диалога с пользователем все ответы и команды пользователя контролируются. Осуществляется автоматическое распознавание и исправление одиночных лексических ошибок и ошибок, связанных с неверным выбором регистра клавиатуры видеотерминала.

16.3.2. Особенности режима решения задач. Для систем, основанных на правилах, решение задач базируется на традиционной схеме интерпретации [Томпсон и др., 1987], включающей циклически повторяющиеся этапы выборки, сопоставления, разрешения конфликтов и выполнения. При этом данные о конкретной проблемной ситуации помещаются в РП. На этапе выборки определяется подмножество активных знаний (правил и объектов РП), которые будут использоваться в текущем цикле работы. На этапе сопоставления определяются значения атрибутов, на которых выполняются условия активных правил. Пара

правило и значения атрибутов, на которых правило выполняется, называется означиванием правила. Результатом работы этапа сопоставления является конфликтный набор, т. е. совокупность означенных правил. На следующем этапе из конфликтного набора выбирается (по определенной стратегии) на исполнение одно означивание. На этапе выполнения исполняется действие выбранного означивания, что приводит к модификации состояния РП.

Реализация стандартной схемы интерпретации в системе ЭКСПЕРТИЗА имеет ряд особенностей, введение которых обусловлено стремлением повысить эффективность решения задач и обеспечить работоспособность системы в сложных проблемных областях.

1. При сохранении способа выполнения правил от данных возможно планирование вывода от целей. На этапе планирования строятся цепочки вывода (применения правил), которые с учетом имеющегося контекста потенциально могут привести к решению задачи. В ходе построения цепочек вывода от целей максимально используется вся имеющаяся в системе информация о состояниях проблемной области с тем, чтобы как можно раньше отвергать цепочки вывода, не способные привести к решению задачи.

2. В проблемных областях, описываемых большим количеством фактов, как правило, не все факты полезны при решении задачи. Присутствие в РП лишних объектов влечет за собой значительную избыточную работу в ходе интерпретации знаний. В системе реализован механизм, позволяющий в начале работы вводить в РП только целевой объект, значения атрибутов которого должны быть вычислены. По мере формирования решения в РП добавляются объекты, важные для получения решения. Определение добавляемых объектов осуществляется по связям, указанным в предпосылке правил, которые еще не отвергнуты. При этом используется механизм наследования значений атрибутов. В ситуациях, когда механизм добавления объектов бесполезен (т. е. все объекты необходимы для решения задач), он может быть отключен. В этом случае до начала решения задачи в РП принудительно должно загружаться полное описание проблемной ситуации.

3. При вычислении значений атрибутов система автоматически манипулирует определенностями, осуществляя их вычисление. Имеется три стандартных способа манипулирования определенностями (максимум, минимум и по формулам системы MYCIN), выбор которых осуществляется установкой значения атрибута управляющего объекта (параметра). Пользователь может ввести свои собственные функции. Стандартно система использует формулы системы MYCIN.

4. Выбор активного набора знаний (правил и объектов) осуществляется в текущем цикле работы системы следующим образом. В начальный момент работы активным является стартовый набор правил, обеспечивающий выбор и загрузку знаний, соответствующих задаче. Управление выбором правил осуществляется с помощью метаправил. Чтобы сократить перебор при поиске решений, правила могут быть разбиты разработчиком ЭС на классы, которые идентифицируются своим именем. Разбивка правил на классы обычно соответствует разбиению задачи на подзадачи или разным способам (моделям) решения одной и той же задачи (подзадачи). Планирование взаимодействия классов реализуется с помощью метаправил перехода.

Механизм выбора активных объектов РП базируется на предположении независимости данных, введенных для разрешения различных подцелей (значений атрибутов объектов), что часто справедливо, особенно для больших проблемных областей. В этом случае данные, введенные в РП по достижении одной цели, могут оказаться ненужными при достижении других целей. Рабочая память динамически структурируется решателем в соответствии с планом вывода. В каждом цикле интерпретации в первую очередь используются объекты, введенные в РП на текущей ветви цепочки вывода (плана). Если возникает необходимость в создании нового объекта (см. выше), то проверяется, нет ли его в других структурах РП, и если есть, то требуемый объект не создается, а просто делается доступным.

Для сильносвязанных проблемных областей, в которых для всех подцелей используются все данные, данный механизм замедляет решение задачи, а для слабосвязанных областей время решения сокращается за счет сокращения затрат на сопоставление. Включение или отключение данного механизма производится установкой соответствующих параметров решателя.

5. Для обеспечения планирования цепочек вывода от целей и работы в условиях неполного представления в РП проблемной ситуации на этапе сопоставления производится определение не только полностью означенных правил, но и частично означенных. Частичным называют означивание, в котором некоторым объектам и атрибутам, перечисленным в левой части правила, не удалось сопоставить соответствующие элементы из РП (потому что такие элементы пока еще не введены в РП). При этом те значения атрибутов, которые сопоставились с левой частью правила, не нарушают выполнимости правила.

Чтобы сократить время работы этапа сопоставления, принимаются следующие меры. Во-первых, конфликтный набор не определяется в каждом цикле заново, а предпринимаются попытки сохранить и использовать в работе конфликтный набор с предыдущего цикла интерпретации. Во-вторых, конфликтный набор определяется и передается на этап разрешения конфликтов не целиком. Зная логику работы этапа разрешения конфликтов, на этапе сопоставления стремятся не получать те означивания, которые будут отклонены этапом разрешения конфликтов. Означивания вычисляются подмножествами так, чтобы не нарушать логику решения задачи и полноту поиска решения, а также сократить перебор, исключив лишние вычисления. Если из первого полученного подмножества ни одно из означиваний выполнить нельзя (в соответствии со стратегией разрешения конфликтов), то вычисляется следующее множество означиваний. Реализовано несколько способов сокращения числа получаемых означиваний, выбор способа осуществляется экспериментально подбором значений параметров решателя.

6. Реализованы два алгоритма разрешения конфликтов. В первом для поддержания одной линии вывода отдается предпочтение означиваниям, использующим значения, которые введены в РП последними. Второй алгоритм отдает предпочтение правилам, являющимся частным случаем других правил. В обоих алгоритмах прежде всего учитывается приоритет правила. Это сделано, чтобы предоставить возможность эксперту влиять на порядок решения задачи. При создании конкретной ЭС можно использовать один из этих алгоритмов.

Дополнительно система ЭКСПЕРТИЗА предоставляет пользователю возможность вмешиваться в работу этапа разрешения конфликтов в процессе решения задачи. Эта возможность реализуется с помощью метаправил управления.

7. В процессе решения задачи система обеспечивает одновременную работу с несколькими альтернативными мирами, получая решение либо в одном из них, наиболее правдоподобном, либо во всех (устанавливается параметром). Альтернативные миры предназначены для представления различных возможных точек зрения на предметную ситуацию. В системе ЭКСПЕРТИЗА альтернативные миры могут быть получены как следствие следующих факторов: противоречивых ответов пользователя, результатов поиска в БД, работы вычислительных функций, применения правдоподобных правил (имеющих коэффициент определенности меньше 1), предположений. Все альтернативные миры хранятся в РП неизбыточным образом в виде дерева. Общие для некоторых миров данные вычисляются один раз. Все достоверные значения являются общими для всех миров. Непротиворечивость миров проверяется метаправилами управления с функцией ПРОТИВОРЕЧИЕ.

16.3.3. Объяснительные возможности системы. В системе реализованы объяснения двух типов, ориентированные на пользователей и на разработчиков ЭС, т. е. на эксперта и инженера по знаниям. Второй тип объяснений является дополнением к первому. Для его использования требуются знания о функционировании системы и минимальные сведения о языке ЛИСП. Объяснения, ориентированные на пользователя, сообщают, для чего предназначена запрашивае-

мая информация и как получены выведенные знания. Объяснения выдаются в виде содержательных ЕЯ-текстов. Смысл объяснений формируется из означенных правил, использованных в выводе. Текст объяснений генерируется подсистемой синтеза текстов на основе информации, содержащейся в описании объектов. Кроме того, пользователю может быть представлен комментарий, описывающий назначение и обоснование целесообразности каждого правила. Эти комментарии составляются разработчиками при создании ЭС.

Объяснения, ориентированные на разработчиков ЭС, предназначены для анализа знаний системы и процесса их использования при решении задачи, не поддерживаются средствами генерации ЕЯ-текстов. Эти объяснения формируются путем интерпретации правил и содержимого РП.

В системе предусмотрены команды, позволяющие задать вопросы о том, какие правила ЭС пыталась применять для решения задач, почему она не смогла их применить, почему некоторое правило в решении задачи не использовалось, где использовалось значение некоторого атрибута и т. д. Основные поддиалоги объяснения системы ЭКСПЕРТИЗА и их назначение приведены в табл. 16.4.

Следует отметить, что в силу неоднозначности причин неприменимости некоторого правила оно может порождать много частичных означиваний и ни разу не выполняться. Объяснения подобных ситуаций могут быть довольно обширны, для их понимания требуется знать содержимое БЗ и методы интерпретации, поэтому пользоваться ими могут либо эксперты, либо инженеры по знаниям (но не конечные пользователи). Объяснения, ориентированные на разработчиков, являются эффективным средством тестирования ЭС и их отладки.

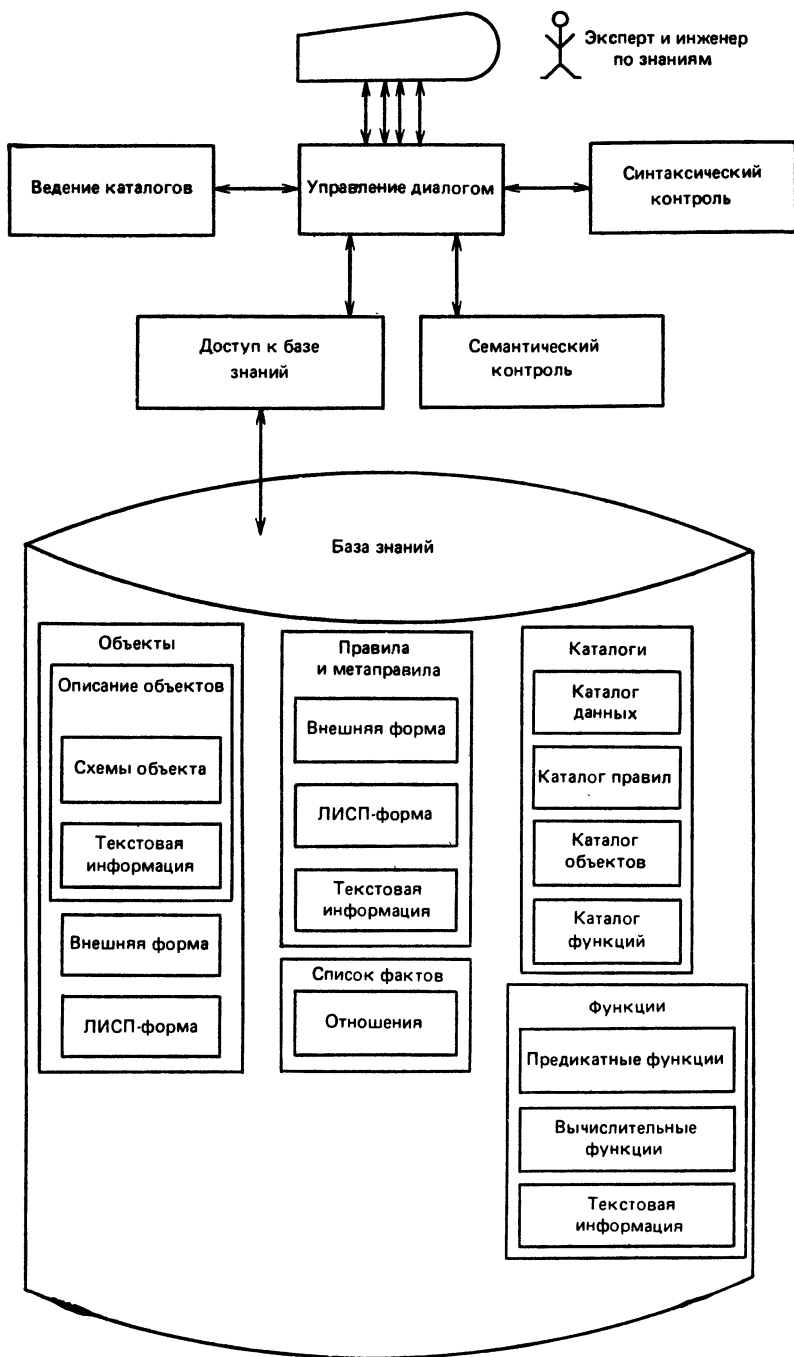
16.4. Функционирование системы

16.4.1. Структура и функционирование средств режима приобретения знаний. Напомним, что структурно система ЭКСПЕРТИЗА выполнена в виде двух независимых подсистем: средств режима приобретения знаний и средств режима ведения консультации. Взаимосвязь подсистем осуществляется через БД. Структура и состав подсистемы режима приобретения знаний представлены на рис. 16.2.

Таблица 16.4

Поддиалоги объяснения

Имя диалога	Комментарий
ПОЧЕМУ КАК ГДЕ	Зачем системе потребовалась запрашиваемая информация Как получено вычисленное значение При вычислении каких значений использовалось указанное значение
СПО	Генерируется список частичных означиваний правила и объясняется, что мешает получить полное означивание
ПОЧЕМУ ПРАВИЛО ТУПИКИ	Почему правило не использовалось при решении задачи Генерируется список атрибутов, значения которых система вычислить не смогла
ПОЧЕМУ ТУПИКИ	Генерируется список правил, которыми пытались вычислить значение атрибута
СПОСОБ	Генерируется список правил, способных вычислить указанный атрибут
МИРЫ	Предоставляется информация о том, какие альтернативные миры и по какой причине созданы



База знаний состоит из основной и вспомогательной частей. Основная часть содержит знания о проблемной области и представлена в виде объектов, правил, метаправил, функций и базы данных. Вспомогательная часть содержит технологическую информацию, предназначенную для разработчиков ЭС и оформленную в виде каталогов.

Часть БЗ, определяющая объекты, содержит описание объектов, внешнюю форму их представления и ЛИСП-форму. Описание объекта оформлено в виде схемы объекта, где указываются имя объекта, входящие в его состав атрибуты и поясняющая их текстовая информация. Внешняя форма содержит сведения о способе отображения объекта на экране видеотерминала. ЛИСП-форма объекта и является описанием объекта на языке ЛИСП, операторы которого интерпретируются подсистемой режима консультации.

Внешняя форма правил и метаправил служит для отображения информации о них пользователю. ЛИСП-форма используется решателем в ходе работы. Текстовая информация правила (метаправила) содержит произвольные сведения о правиле и причинах его появления в системе. Данная информация может выдаваться пользователю при формировании объяснений.

Предикатные и вычислительные функции написаны на языке ЛИСП, могут быть снабжены поясняющей текстовой информацией. База данных состоит из совокупности поименованных отношений, каждое отношение записывается на языке ЛИСП.

Технологически для хранения знаний используются библиотеки ОС ЕС ЭВМ и стандартный библиотечный метод доступа. Все компоненты БЗ (рис. 16.2) хранятся в виде одного или нескольких разделов библиотек. Информация о том, в каком разделе какие знания хранятся, собрана в каталоги. Кроме того, в каталогах содержится информация о времени создания знаний и об их авторах. Каталоги позволяют систематически просматривать знания, принадлежащие ЭС, по имени и типу синтаксической конструкции определять раздел, в котором она находится.

Программные средства режима приобретения знаний состоят из средств управления диалогом, которые в соответствии с заданными сценариями (глобальной структурой диалога приобретения знаний) обеспечивают сопряженный и корректный ввод информации в БЗ. В процессе ввода информации средства управления диалогом выполняют следующие функции:

- синтаксический контроль (обеспечивает контроль каждого сообщения, вводимого пользователем);

- семантический контроль (обеспечивает проверку на непротиворечивость вводимой пользователем информации с содержанием базы знаний и построение отчета о БЗ, позволяющего осуществлять ее анализ);

- доступ к базе знаний (обеспечивает запись и выборку информации из БЗ, хранящейся на внешнем носителе);

- ведение каталогов (обеспечивает автоматическое поддержание и использование технологической информации).

Механизм приобретения знаний основывается на применении шаблонов синтаксических конструкций языка представления знаний, выдаваемых пользователю. В качестве шаблона пользователю разрешается использовать любую находящуюся в БЗ информацию (описание объекта, правило и т. д.). Разрешается перенос информации из уже существующих элементов БЗ во вновь создаваемые.

Средства управления диалогом для удобства работы пользователя обеспечивают параллельно с вводом информации ведение дополнительно до трех диалогов, позволяющих просматривать информацию, содержащуюся в БЗ, с целью ее анализа и использования при вводе.

16.4.2. Структура и функционирование подсистемы режима ведения консультации. Подсистема режима консультации предназначена для решения задач, работает и при создании, и при использовании ЭС. Структура и состав средств приведены на рис. 16.3. Управление консультацией осуществляется в соответствии со структурой метадIALOGA, правилами и метаправилами и командами

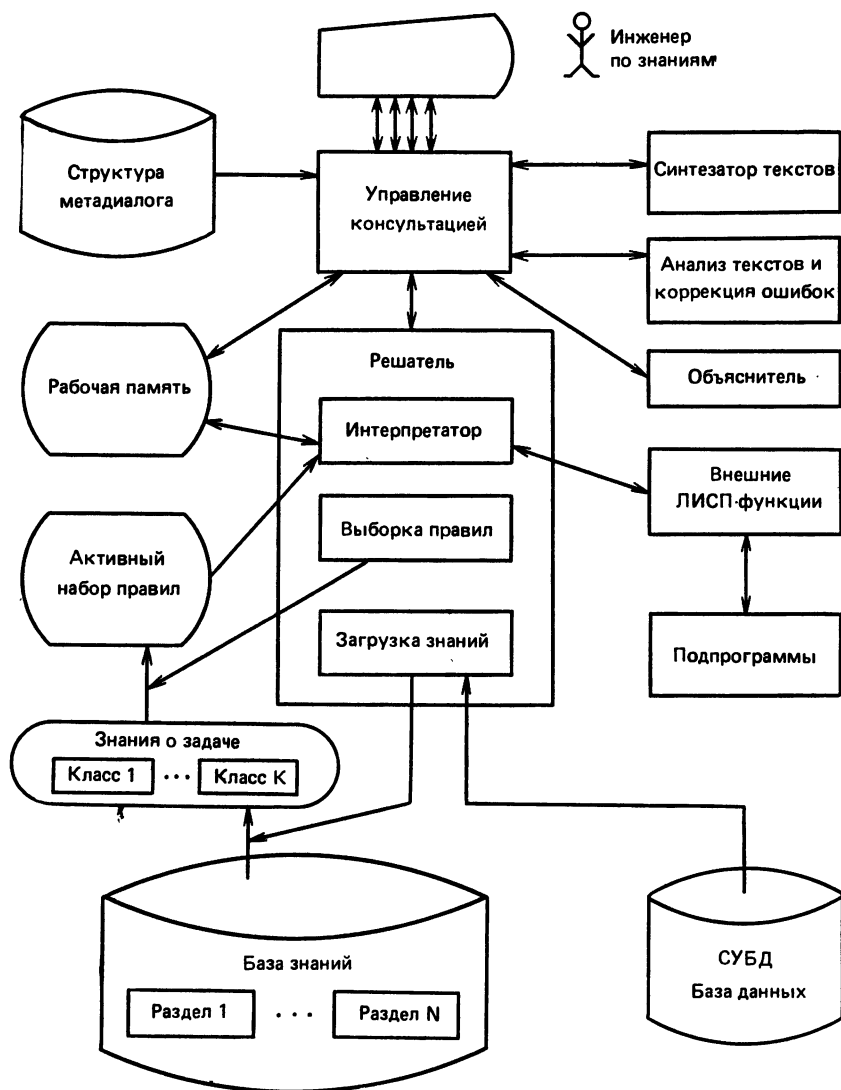


Рис. 16.3. Средства ведения консультации системы ЭКСПЕРТИЗА

пользователя. Средства управления консультацией реализуют структуру метадиалога, выполняя следующие функции: решение задач; объяснение функционирования системы; синтез ЕЯ-сообщений системы; анализ ответов пользователя, контроль и коррекция ошибок.

В решателе интерпретатор знаний непосредственно вычисляет значения атрибутов на основе содержимого РП, активного набора правил и функций. Выборка правил осуществляет на основе метаправил загрузку активного набора правил. Загрузка знаний помещает в оперативную память подмножество знаний БЗ, имеющих отношение к решаемой задаче, и структурирует их по классам в соответствии с метаправилами загрузки знаний.

Рабочая память, активный набор правил и знания о задаче являются информационными компонентами, хранящимися в течение консультации в оперативной памяти ЭВМ. Все программные средства, вызываемые средствами управления консультацией, используют знания, находящиеся в информационных компонентах. Факты из внешней БД попадают в РП через интерпретатор, так же как значения атрибутов, полученные из внешних функций. Факты, полученные от пользователя, помещаются в РП непосредственно средствами управления консультацией.

16.4.3. Реализация и использование. Система ЭКСПЕРТИЗА реализована на языках программирования ЛИСП ВМ и АСЕМБЛЕР ОС ЕС ЭВМ. Общий объем программного обеспечения, включающий технологические средства, около 25 тыс. строк текста на языке ЛИСП и 8 тыс. команд на языке АСЕМБЛЕР. В разработке системы участвовало от 4 до 10 человек. Общая трудоемкость работ по программированию, отладке и выпуску документации системы около 25 чел.-лет.

Система функционирует на ЕС ЭВМ под управлением ОС ЕС 6.1 (режимы MFT с подзадачами или MVT) или ОС 7.0. Функционирование системы требует как минимум 800 Кбайт оперативной памяти (увеличение емкости выделенной памяти приводит к повышению скорости решения задач) и 10 Мбайт внешней памяти (на устройствах прямого доступа). Связь с пользователем осуществляется через дисплей типа ЕС-7920. С помощью системы ЭКСПЕРТИЗА создан ряд промышленных и исследовательских прототипов ЭС.

В 1989 г. заверены следующие работы по развитию ИС ЭКСПЕРТИЗА (ГЛОБ): перенос системы на ПЭВМ типа IBM PC/AT с сопроцессором Intel 80386 и на ПЭВМ PS/2 (с памятью более 2 Мбайт); автоматическое построение скомпилированной базы знаний для системы ЭКСПЕРТ (см. § 15.2) из базы знаний системы ЭКСПЕРТИЗА, что позволяет на основе общих знаний системы ЭКСПЕРТИЗА строить конкретные эффективные системы для работы на ПЭВМ с ограниченной вычислительной мощностью.

Глава 17.

Отечественные экспертные системы на персональных ЭВМ

17.1. Система СПЭИС/МОДИС

О. В. Ковригин

17.1.1. Постановка задачи. Целью исследований было создание ЭС диагностики различных форм симптоматической гипертензии МОДИС-2. Для построения этой ЭС использовалось ИС СПЭИС (см. § 15.1). Актуальность решаемой задачи обусловлена тем, что гипертензией страдает 10% населения Земли. Причины повышения артериального давления очень разнообразны — более 30 основных заболеваний. Сложность диагностики этих заболеваний состоит в том, что они могут относиться к компетенции специалистов из различных областей медици-

ны: нефрологии, ангиологии, урологии и т. п. Таких специалистов в настоящее время явно не хватает. Создание ЭС, которая бы содержала знания лучших специалистов по различным областям медицины, связанных с гипертензией, позволяло бы оказать существенную помощь в повышении уровня диагностики указанных патологий.

Экспертная система МОДИС-2 имеет важную особенность. Ее использование предполагается как в поликлиниках общего профиля, так и в специализированных учреждениях. Очевидно, что уровень доступной информации о больном в этих заведениях может быть разным. В поликлинике терапевту доступна информация более общего характера: жалобы пациента, данные внешнего осмотра, история болезни и результаты общих анализов. В этом случае ЭС должна дать рекомендации, к каким специалистам следует обратиться, направить на специальные исследования. Таким образом, на основе имеющейся информации общего характера ЭС должна сузить круг подозреваемых заболеваний, выбрать наиболее вероятные. При использовании ЭС в специализированном учреждении может быть доступна более детальная информация о больном — данные специальных исследований (например, результаты проведенной аортографии или внутривенной урографии). В этом случае ЭС должна ставить по возможности точный диагноз.

Такая постановка задачи позволила обратить внимание на три важных момента при разработке ЭС: ЭС должна содержать знания различных специалистов и, следовательно, может рассматриваться как совокупность отдельных ЭС; ЭС должна адаптироваться к уровню доступной информации и быть в состоянии принимать решения в условиях отсутствия части данных о больном; профессиональный уровень потенциальных пользователей ЭС может быть крайне разнообразен — от терапевта районной поликлиники до специалиста в области гипертензии.

Второй этап создания ЭС предполагает определение основных концепций проблемной области — выделение основных объектов и понятий. Это, на наш взгляд, один из важнейших этапов в создании ЭС, так как позволяет определить, какого характера знаниями оперирует эксперт для описания основных концепций — проблемной области.

Анализ показал, что основными концепциями, с которыми работает эксперт при решении задачи, являются заболевания, их формы и группы, симптомы, данные лабораторных исследований и т. п. Для их описания он использует знания трех видов (конечно, такое разбиение знаний на три вида условно, но будет полезно в дальнейшем для формирования структур представления).

1. Описательные знания, которые используются экспертом для описания отношений между понятиями. Например, такое понятие, как систолическое давление, связано у эксперта с дополнительной информацией. Так, он знает, что систолическое давление — показатель состояния больного, измеряется некоторым числом, которое может находиться в пределах от 50 до 350, знает также, что систолическое давление всегда больше диастолического и т. п. К описательным знаниям можно отнести и знания эксперта о заболеваниях и их иерархии. Например, знания эксперта о том, что гломерулонефрит — это заболевание почек, относящееся к нефрологической группе заболеваний, и может иметь острую, подострую или хроническую формы.

2. Знания о том, как эксперт выводит новые утверждения на основе анализа некоторой информации. Эксперт пользуется этими знаниями для постановки диагнозов, определения синдромов и т. п. Характерным для этого вида знаний можно считать диагностическое правило «Если больной молодого возраста и у него наблюдаются пароксизмы давления и результаты гормональных исследований крови или мочи показывают увеличенное более чем в 2 раза содержание катехоламинов, то у больного, по всей вероятности, феохромоцитома».

3. Знания о том, как эксперт выбирает наиболее перспективные пути поиска решений. Такие знания, представленные в виде правил, помогают эксперту не рассматривать маловероятные, с его точки зрения, гипотезы о заболеваниях и подключать к решению задачи наиболее релевантные знания. Это зна-

ния метауровня, так как они определяют некоторую схему использования знаний первого и второго видов. Примером такого правила можно считать: «Если у больного имеются значительные изменения в анализах мочи, то прежде всего следует подозревать нефрологическую природу гипертонии».

На втором этапе проектирования ЭС производится также попытка выявить общую схему вывода решений, которой пользуется эксперт при диагностике заболеваний. Сначала эксперт собирает некоторую начальную информацию о больном. Врач выслушивает жалобы, задает вопросы о перенесенных заболеваниях, уточняет некоторые неясные моменты, осматривает пациента. После этого на основе анализа информации, собранной на начальном этапе, он выдвигает гипотезы, как правило, общего характера. Врач пытается определить наиболее вероятный общий характер гипертонии (нефрологической, сосудистой или какой-либо другой). Затем, установив общий характер заболевания, он пытается поставить более точный диагноз. Можно считать, что генерация гипотез производится по схеме от общего к частному. Выдвижение гипотез сопровождается их проверкой путем сбора и анализа дополнительной информации. Новые данные о больном могут либо подтвердить гипотезу, либо опровергнуть ее.

Однако общий порядок генерации гипотез может нарушаться, если в самом начале эксперт обладает настолько специфической информацией, что она позволяет ему сгенерировать некоторую частную гипотезу, минуя общие. Подобная схема поиска решения, где каждая гипотеза выдвигается только в определенных условиях, позволяет значительно сократить число возможных альтернатив решения.

Разработанный в МОДИС-2 язык представления знаний позволяет представлять все три вида указанных знаний, а решатель обеспечивает процедуру поиска решения, которая близка тому, как это делает эксперт.

17.1.2. Представление знаний. Выделенные типы экспертных знаний требуют для их представления конструкции различных видов. Знания второго и третьего видов выражаются правилами и могут представляться с помощью продукций.

В МОДИС-2 условия применимости записываются в форме, близкой к естественному языку: «Если ХАРАКТЕР ПОВЫШЕНИЯ АД=ПАРАКСИЗМАЛЬНЫЙ», или «Если И (ГИПОТЕЗЫ ПРОВЕРЕНЫ Г₁Г₂Г₃) (НЕ ОПРЕДЕЛЕНО РЕШЕНИЕ Д₁Д₂Д₃)», или «Если ВОЗРАСТ МЕНЕЕ 40».

Совокупность необходимых предикативных символов была определена исходя из анализа проблемной области. Предикаты в левой части правила могут соединяться логическими операциями конъюнкции, дизъюнкции и отрицания. Для записи правых частей правил удобно воспользоваться процедурами. Так, правило с правой частью вида «..., то у больного феохромоцитоме» можно представить в виде процедуры (СДЕЛАТЬ ЗАКЛЮЧЕНИЕ: «У БОЛЬНОГО ФЕОХРОМОЦИТОМА»), а правило с правой частью вида «..., то рассматривать нефрологическую группу заболеваний» — в виде (АКТИВИЗИРОВАТЬ: НЕФРОЛОГИЧЕСКУЮ ГРУППУ).

Полный набор необходимых процедур около 20 наименований также был определен исходя из анализа высказываний эксперта. Однако он может быть легко расширен путем добавления новых в соответствующую библиотеку.

Итак, для представления правил используется общепринятая форма продукционных правил. Однако в отличие от «чистых» продукционных систем в МОДИС-2 БЗ не представляет собой простую совокупность продукций.

Если проанализировать всю совокупность правил, которые эксперт в состоянии сформулировать для рассматриваемой области знаний, то можно обратить внимание, что среди этого множества выделяются группы правил, которые в своих правых частях делают различные заключения об одних и тех же понятиях или объектах. Объединим такие правила в рамках отдельных структур, имена которых будут соответствовать понятиям или объектам, упоминающимся в правых частях объединенных продукций. Тогда все множество продукций окажется распределенным по отдельным структурам, и БЗ будет представлять собой совокупность относительно крупных структур.

Принцип распределенного представления знаний можно распространить и на эвристические правила поиска решений. В этом случае правила группируются в рамках структур, которые представляют некоторые опорные ситуации. В этих опорных ситуациях должен быть сделан альтернативный выбор между направлениями поиска решения.

Среди описательных знаний эксперта также можно выделить «сегменты» знаний, которые используются для характеристики отдельных понятий и объектов. Таким образом, к каждой семантической единице проблемной области могут быть присоединены знания как описательного, так и процедурного вида. Такое распределенное представление знаний позволяет организовать четкую структуризацию БЗ. Это значительно облегчает процесс создания БЗ и ее понимание, а кроме того, такая структуризация позволяет организовать простой и эффективный способ вывода решений.

В МОДИС-2 структуры, используемые для представления знаний, называют концепциями. Они могут быть двух видов: Н- и Р-концепции (см. § 15.1). Р-концепции используются для представления знаний о диагностических признаках (симптомы, данные лабораторных исследований). Отметим, что представляемые в БЗ проблемно-ориентированной ЭС параметры образуют фрагменты связанных вершин. Каждая такая связь имеет свою интерпретацию, которая описывается на этапе проектирования языка представления знаний и добавляется к библиотеке функций интерпретации концепций. Обращение к знаниям об одном из параметров при решении задач ЭС влечет за собой активизацию знаний о других параметрах, находящихся с первым в одном связанном фрагменте. Это позволяет эксперту, один раз полностью описав параметры и их взаимосвязи, оперировать ими далее в процессе формирования диагностических и эвристических правил, не вдаваясь в тонкости их взаимосвязей, использовать агрегированные сегменты знаний. С помощью Н-концепций представляются знания о заболеваниях и их группах. Кроме того, Н-концепции используются для записи стратегий поиска решения.

Язык Н- и Р-концепций позволяет полностью описать знания эксперта о предметной области и эвристические знания о том, как эксперт ищет решение. Интересно отметить, что в процессе построения БЗ неоднократно приходилось возвращаться к третьему этапу, внося коррективы в язык представления знаний (ЯПЗ) для усиления его выразительных возможностей. Такие исправления были бы крайне затруднены без средств проектирования СПЭИС.

Процесс заполнения БЗ в МОДИС-2 производится в два этапа. На первом описывается все пространство параметров проблемной области. Каждый параметр представляется Р-концепцией. На этом этапе формируются связанные фрагменты параметров. С помощью редактора БЗ, реализованного в СПЭИС, возможно заполнение концепций в диалоговом режиме, когда специальная программа предлагает заполнить слоты на языке, близком к естественному, а затем сама переводит их в требуемую форму. Этот редактор позволяет также сформировать графическое представление связанных фрагментов, что облегчает анализ связей между параметрами в БЗ. Полностью описанные параметры составляют нижний уровень знаний эксперта в БЗ.

На втором этапе заполнения БЗ формируются два других уровня знаний. Сначала, исходя из перечня заболеваний, которые должна диагностировать ЭС, строятся «мини-эксперты» для каждого заболевания, их групп и форм, оформляемые в виде Н-концепций. Такие «мини-эксперты» образуют второй уровень знаний эксперта в БЗ. И в заключение формируются Н-концепции для метазнаний эксперта. Фактически на этом этапе производится формирование правил (в рамках Н-концепций), управляющих использованием знаний второго уровня. Концепции, в которых представлены метазнания, образуют третий уровень знаний в БЗ. Эвристические метазнания могут добавляться и к знаниям второго уровня, что позволяет «мини-экспертам» активизировать друг друга без помощи «метаэкспертов».

Таким образом, БЗ системы МОДИС-2 представляется в виде совокупности концепций, распределенных по трем уровням, причем концепции каждого

уровня могут быть связаны как с концепциями своего уровня, так и с концепциями других уровней.

17.1.3. Решатель. Чтобы работа ЭС имитировала поведение эксперта при решении диагностических задач и чтобы ее функционирование было понятно эксперту, необходимо при создании ЭС стремиться к использованию такого решателя, который бы наиболее полно позволял моделировать процесс принятия решений экспертом.

Пользуясь ИС СПЭИС, в решателе проектируемой ЭС МОДИС-2 удалось воплотить такую схему вывода, которая позволяла организовать поведение ЭС, близкое к поведению эксперта. Этому способствовали три важных обстоятельства: организация БЗ в виде отдельных концепций с распределенными по ним системами продукций; представление эвристик эксперта о том, как он ищет решение, в виде систем продукций, также распределенных по отдельным концепциям; возможность использования процедур в ЯПЗ.

Действительно, для каждого отдельного «мини-эксперта», представленного Н-концепцией, знания эксперта, находящиеся внутри, полностью определяют его поведение, т. е. операции, которые он должен выполнить. Ведь в слоте BODY Н-концепции находятся конструкции двух видов. Конструкция первого вида соответствует процедурам определения необходимой для этого «эксперта» информации, а конструкция второго вида — продукциям, истинность которых приводит к выполнению определенного ряда процедур, таких как: СДЕЛАТЬ ЗАКЛЮЧЕНИЕ; АКТИВИЗИРОВАТЬ; и т. д. Значит, запись знаний эксперта в Н-концепцию формирует ее поведение. Для простоты условимся, что записанная в этом слоте информация оценивается последовательно. Тогда в процессе работы «эксперт» для сбора информации будет обращаться к первому уровню знаний, подключая необходимые Р-концепции для определения значений параметров, в которых нуждаются продукции «эксперта». Оценивая свои продукции, «эксперт» будет генерировать или некоторые утверждения, или команды на передачу управления другому «мини-эксперту» или «метаэксперту».

Чтобы получаемая в процессе вывода информация была доступна всем концепциям БЗ, выделяется определенная область памяти, открытая всем концепциям БЗ. Такая память подразделяется на две части. Первая служит для хранения установленных параметров большого и утверждений, формируемых «экспертами». Эту часть можно сравнить с глобальной базой данных в продукционных системах. Информация, записанная в ней, определяет условия активизации «экспертов». В нее же поступают новые сгенерированные утверждения, что может привести к активизации новых «мини-экспертов» или «метаэкспертов». Вторая часть памяти (список работ) содержит имена «мини- и метаэкспертов», которые были рекомендованы к активизации. «Эксперты» ставятся в очередь. После завершения рассмотрения очередного «эксперта» он удаляется из очереди, а первый из очереди направляется на рассмотрение.

Постановка Н-концепции в очередь производится продукциями или процедурами, содержащимися в «мини- и метаэкспертах». Таким образом, решатель просто интерпретирует информацию слотов Н-концепций, что приводит к тому, что процесс активизации гипотез (а следовательно, и процесс поиска решений) строится согласно правилам, сформулированным экспертом.

Поиск решения ЭС всегда начинается с одного определенного «метаэксперта», а затем, исходя из условий текущей задачи, активизируются все новые Н-концепции второго и третьего уровней БЗ. Они, в свою очередь, постоянно обращаются к первому уровню (Р-концепциям), с которого тоже может произойти добавление новой Н-концепции к списку работ.

Обращение к нижнему уровню происходит, если Н-концепции требуется определить значение параметра согласно процедуре (ОПРЕДЕЛИТЬ: Р₁) или если значение этого параметра пока отсутствует в доступной памяти. Активируя соответствующую параметру Р-концепцию, решатель оценивает информацию, записанную в ней. В результате будет либо порождено дерево вывода на связанном фрагменте параметров и значение искомого параметра будет опре-

делено согласно значениям других параметров, либо его значение будет запрошено у пользователя. После установления значения параметра активизируются все связи параметра. Обращение к одной Р-концепции нижнего уровня может активизировать довольно сложную цепную реакцию в пространстве Р-концепции, результатами которой могут быть как требования установить значения новым параметрам, так и добавление гипотез в список работ.

Анализ Н-концепций состоит в последовательной интерпретации записей структуры. Концепция считается проанализированной до конца, если все записи проинтерпретированы. Однако иногда необходимо остановить процесс интерпретации и перейти к анализу другой Н-концепции. Для этих целей используется процедура СТОП., которая прекращает анализ концепции и передает управление решателю для активизации следующей по очереди концепции. Выполнив ряд работ, решатель возвращает управление на прерванную концепцию, и ее анализ будет продолжен с записи, следующей за процедурой СТОП.:

Чтобы обеспечить ЭС способность адаптироваться к различным уровням доступной информации (свойство, о котором упоминалось на этапе постановки задачи), в Н-концепциях разрешено использовать продукции с различными весами. Пусть заключения о понятии, описываемом концепцией, содержащиеся в правой части продукций, имеют некоторый вес, обозначаемый целым числом, и в процессе оценки Н-концепции выполняется некоторая такая продукция. Тогда, если анализ концепции не закончен, заключение, сделанное этой продукцией, не будет занесено сразу в общую память. Если при дальнейшем анализе концепции выполняется другая продукция, имеющая некоторый вес заключения, то вес последнего заключения будет сопоставлен с весом уже сделанного ранее. В результате будет выбрано заключение с большим весом. Такое сопоставление заключений будет производиться до окончания анализа Н-концепции. Следовательно, после завершения работы источника знаний в общую память будет направлено заключение с наибольшим весом. Такая организация заключений в продукциях позволяет организовать поиск сначала наиболее грубого заключения, уточняя его, если появится необходимая информация.

Отметим, что БЗ для решения задачи диагностики симптоматической гипертонии содержит более 250 концепций и около 500 правил. Постоянная модификация БЗ выявила значительные преимущества предложенного ЯПЗ. Последствия вносимых в концепции изменений легко проследить как в рамках самой концепции, так и вне ее.

17.1.4. Функционирование системы. Работа МОДИС-2 начинается со сбора начальной информации, куда входят анкетные данные больного, его жалобы, история болезни, симптомы и лабораторные анализы. На экране дисплея, перед которым сидит пользователь, появляются вопросы и списки возможных ответов. Пользователь должен выбрать из предложенного списка подходящий ответ на заданный вопрос. Если вопрос или предлагаемые ответы не понятны пользователю, он может затребовать дополнительную информацию. Ответы, вводимые пользователем в ЭВМ, проверяются программой, и, если обнаруживаются противоречия в ответах, ЭС сигнализирует об этом. В системе МОДИС-2 реализован дружественный интерфейс общения с пользователем, поддерживающий ввод и вывод информации в удобном виде (многооконные режимы, графические изображения и т. п.).

На этапе сбора начальной информации система МОДИС-2 задает около 30 вопросов. Потом задаются пользователю вопросы для анализа имеющихся гипотез. Выдвигая, подтверждая и отвергая гипотезы, система МОДИС-2 ищет решение. Время одного диагностического сеанса и число задаваемых вопросов зависят от сложности решаемой задачи. В среднем длительность одного диагностического сеанса 5—10 мин. В результате МОДИС-2 выдает или точное заключение о причине повышения давления, или диагноз, подлежащий дальнейшему уточнению, или рекомендации на проведение целенаправленных исследований. Окончательное заключение, даваемое системой, является совокупностью элементарных решений по отдельным гипотезам и формируется в процессе анализа дерева поиска решений. Характерным заключением МОДИС-2 можно счи-

тать такое: «У больного, по всей видимости, феохромоцитомы. Он нуждается в проведении гормональных исследований крови и мочи с целью определения содержания там катехоламинов. Больному необходима консультация нефролога. Для исключения синдрома Кона необходимо проведение биохимических исследований».

В системе МОДИС-2 реализован довольно широкий спектр возможностей для объяснения как процесса поиска решения, так и получаемых заключений. Прежде всего, процесс активизации гипотез и их проверки сопровождается сообщениями системы о том, какая концепция находится на рассмотрении, кроме того, пользователь информируется о результатах проверки гипотез, их подтверждении или отклонении. Это позволяет следить за «ходом» рассуждений системы. В процессе вывода решений или после получения решения эксперт или пользователь может обратиться к системе с запросом о том, почему было получено такое решение (либо конечное, либо промежуточное). В этом случае МОДИС-2 анализирует дерево поиска решений, демонстрирует пользователю цепочку вывода, обосновывая каждый шаг вывода, полученный в результате применения определенных «сегментов» знаний к данным о пациенте. На этом этапе используются программные средства поддержки ретроспективных и встроенных объяснений, реализованных в СПЭИС.

Гипотетические объяснения, обеспечиваемые системой МОДИС-2, позволяют эксперту знакомиться с содержанием БЗ, анализировать различные альтернативы возбуждения концепций. Информация, сообщаемая системой в ответ на эти запросы, позволяет эксперту определить недостатки знаний ЭС и исправить их, модифицировав необходимые правила.

17.2. Система ОБЕЗБОЛИВАНИЕ/ЭКСПЕРТ

Б. С. Кирсанов, М. Д. Шапот

17.2.1. Постановка задачи. Обезболивание является одной из актуальных проблем стоматологии. Особенно широко применяется местное обезболивание, что обусловлено его относительной безопасностью и быстротой выполнения. Для правильного выполнения анестезии врач-стоматолог должен хорошо представлять топографию нервных стволов и окружающих их клетчаточных пространств, а также учитывать вид и объем предполагаемого вмешательства, возрастные особенности и другие факторы. Актуальной и важной проблемой является выбор адекватного способа обезболивания при наличии сопутствующих заболеваний, таких как сердечно-сосудистые патологии и др. Выбирая методы и средства местного обезболивания при проведении амбулаторных стоматологических вмешательств, специалисты широко используют знания эвристического характера, учитывают многие факторы. В связи с этим можно считать, что одним из эффективных способов обучения студентов, а также ознакомления их и врачей-стоматологов с новейшими достижениями в рассматриваемой области является использование ЭС. Эта система должна демонстрировать рассуждения специалиста при выборе методов и средств местной анестезии в различных ситуациях, а также содержать справочную информацию о технике проведения обезболивания, применяемых лекарственных средствах, возможных осложнениях и ошибках. Пользователями системы могут быть как студенты медицинских институтов, так и врачи — слушатели курсов повышения квалификации.

В настоящее время разработан исследовательский прототип ЭС, используемый в учебном процессе для обучения выбору методов и средств обезболивания, а также методам проектирования ЭС. Система создана на основе ИС ЭКСПЕРТ (см. § 15.2) и функционирует на ПЭВМ ЕС-1840 или IBM PC (XT/AT).

Местная анестезия в челюстно-лицевой области включает инъекционные и неинъекционные методы. К инъекционным относятся инфилтративная и проводниковая анестезия. Инфилтративная анестезия заключается в непосредственном пропитывании операционного поля обезболивающим раствором. В систе-

ме различается инфильтрационная анестезия мягких тканей лица, на верхней челюсти, на нижней челюсти, поднадкостничная и внутрикостная анестезия. При проводниковой анестезии обезболивающий раствор вводится не в место, подлежащее оперативному вмешательству, а в доступно лежащую часть нерва, иннервирующего всю данную область. К проводниковой анестезии относятся: туберальная (блокада задних верхних альвеолярных нервов), подглазничная (блокада передних и средних верхних альвеолярных нервов у подглазничного отверстия) и др. Из неинъекционных методов в исследовательском прототипе представлена аппликационная (поверхностная) анестезия, применяемая при малотравматичных вмешательствах на поверхностно расположенных участках мягких тканей полости рта и лица.

Обезболивание достигается блокированием всех нервов, иннервирующих участок предполагаемого вмешательства. При определении группы нервов, подлежащих обезболиванию, необходимо учитывать возможные особенности расположения нервов у различных людей. Как правило, каждый нерв может быть блокирован несколькими способами (для его анестезии можно блокировать тот нерв, ветвью которого он является, и т. д.). При выборе метода обезболивания нерва учитывается ряд факторов, и в первую очередь доступность нерва. Например, нецелесообразно применять инфильтрационную анестезию в случае расположения нерва в толще челюсти. К другим факторам можно отнести: сложность метода, опасность осложнений, риск в случае ошибки, наличие воспалительных процессов на челюстях, требуемое количество анестетика, возраст пациента и др.

При проведении местного обезболивания большое значение имеет правильный выбор анестетика, его сочетание с вазоконстриктором, их дозы и концентрации. В ЭС описаны следующие лекарственные средства: новокаин, тримекаин, лидокаин, бупивакаин, пиромекаин, вазоконстрикторы адреналин и норадреналин¹.

17.2.2. Функционирование ЭС. Рассмотрим порядок проведения консультации ЭС. В ходе консультации ЭС решает следующие подзадачи (рис. 17.1).

В начале консультации решается вопрос о принципиальной возможности проведения местного обезболивания. В случае обнаружения противопоказаний пользователю выдаются сообщения о них, и работа заканчивается. Если противопоказания не обнаружены, система переходит к вводу исходных данных: возраста пациента, информации об основном и сопутствующих заболеваниях, локализации, виде и объеме предполагаемого вмешательства.

На следующем этапе работы система определяет, какие нервы и с какой степенью определенности могут иннервировать область, подлежащую вмешательству. Дальнейшие действия системы определяются той задачей, которую хочет решать пользователь. Если целью консультации является анализ ошибок при проведении местного обезболивания, то система запрашивает у пользователя, как и с помощью каких лекарственных средств осуществлялось обезболивание, после чего переходит к решению задачи анализа ошибок. Если же консультация проводится для выбора методов и средств обезболивания, то система определит для каждого подлежащего блокированию нерва возможные в данной ситуации методы обезболивания и сообщит о них пользователю в порядке, соответствующем целесообразности их применения. Пользователь может получить на данном этапе справочную информацию о каждом рекомендуемом методе, после чего он должен выбрать способ обезболивания, т. е. указать, каким образом он хочет блокировать каждый отключаемый нерв. Система выводит

¹ Более подробные сведения о постановке задачи можно найти в

Егоров П. М. Местное обезболивание в стоматологии — М.: Медицина, 1985.
Кузин М. И., Харнас С. Ш. Местное обезболивание. — М. Медицина, 1982.
Шугайлов И. А. Разработка метода оценки болевой чувствительности и эффективности местного обезболивания в стоматологии// Стоматология. — 1984. — № 6. — С. 63—66.

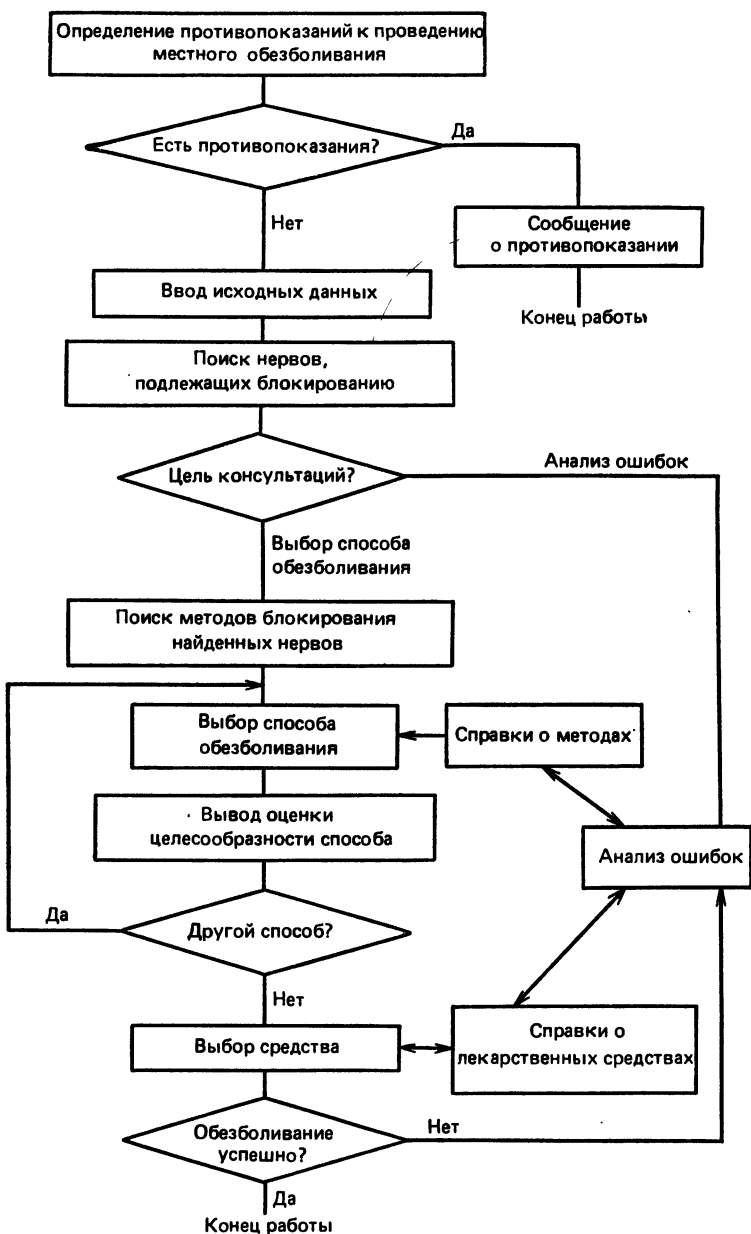


Рис. 17.1. Подзадачи экспертной системы по выбору методов и средств местного обезболивания

Сценарий управляющей подсистемы

№ п/п	Предложение сценария
1	ЕСЛИ [МЕСТНОЕ _ ОБЕЗБОЛИВАНИЕ. возможно] < 10.0 ТО СТОП «Выявлены противопоказания к проведению местного обез-
2	ЦЕЛЬ ЗАБОЛЕВАНИЕ, ВИД _ ВМЕШАТЕЛЬСТВА, ЛОКАЛИЗАЦИЯ
3	РЕЗУЛЬТАТ НЕРВ
4	ЕСЛИ анализ ТО ПЕРЕХОД 17
5	ЦЕЛЬ МЕТОД
6	РЕШИТЬ ВЫБОР _М
7	РЕЗУЛЬТАТ [ПРИМЕНЕНИЕ. целесообразно]
8	ЕСЛИ ~ повторить ТО ПЕРЕХОД 11
9	СБРОС МЕТОД, повторить
10	ПЕРЕХОД 5
11	РЕЗУЛЬТАТ СРЕДСТВО
12	СБРОС СРЕДСТВО
13	РЕШИТЬ ВЫБОР _С
14	РЕЗУЛЬТАТ ДОЗА _АНЕСТЕТИКА, КОНЦЕНТРАЦИЯ _АНЕСТЕТИКА, ДОЗА _ВАЗОКОНСТРИКТОРА, КОНЦЕНТРАЦИЯ _ВАЗОКОНСТРИКТОРА
15	ЕСЛИ удачно ТО СТОП
16	ПЕРЕХОД 20
17	РЕШИТЬ ВЫБОР _М
18	РЕШИТЬ ВЫБОР _С
19	РЕШИТЬ ВЫБОР _Д
20	РЕШИТЬ АНАЛИЗ

оценку целесообразности применения выбранного способа, учитывая такие факторы, как количество уколов, трудоемкость, риск, количество анестетика, продолжительность действия и др. Пользователь может изменить свое решение, выбрать другой способ и получить оценку его целесообразности. После принятия пользователем окончательного решения система переходит к выбору средств обезболивания. Пользователю сообщается о всех возможных в данном случае лекарственных средствах, их дозах и концентрациях, а также выводятся оценки целесообразности применения этих средств. При оценке целесообразности учитываются такие факторы, как переносимость данного анестетика, анестезирующее действие, токсичность и др. Пользователь может ознакомиться со справочной информацией о любом из рекомендуемых препаратов. Если обезболивание прошло неудачно, система переходит к решению задачи анализа ошибок.

17.2.3. Представление знаний. Знания хранятся в базе знаний. Они представляют собой совокупность взаимосвязанных подсистем, решающих отдельные подзадачи (например, выявление противопоказаний). Управление консультацией, а также глобальные атрибуты проблемной области (т. е. используемые всеми подсистемами) описываются в управляющей подсистеме. Целевыми глобальными атрибутами являются: МЕТОД (значения: мандибулярная _слева, мандибулярная справа, туберальная _слева и др.), ПРИМЕНЕНИЕ (значения: целесообразно, трудоемко, имеет _риск, опасно _осложнениями и др.), СРЕДСТВО (новокаин, тримекаин и др.), числовые атрибуты ДОЗА _АНЕСТЕТИКА, КОНЦЕНТРАЦИЯ _АНЕСТЕТИКА, ДОЗА _ВАЗОКОНСТРИКТОРА, КОНЦЕНТРАЦИЯ _ВАЗОКОНСТРИКТОРА. Сценарий управляющей подсистемы приведен в табл. 17.1.

Первая подзадача, решаемая ЭС, — поиск возможных противопоказаний к проведению местного обезболивания. В качестве примера рассмотрим реализацию этой подсистемы в исследовательском прототипе. В данном случае подзадача рассматривается упрощенно, считается, что противопоказаниями являются: непереносимость местных анестетиков, органические нарушения центральной нервной системы (олигофрения, болезнь Дауна), тяжелые формы гиперкинезов, большие вмешательства. В данной подсистеме определены следующие атрибуты: ПРОТИВОПОКАЗАНИЯ (значения — аллергия, нарушения ц-н-с, гиперкинез, большое-вмешательство), ВЫЗЫВАЕТ_АЛЛЕРГИЮ (значения — все описанные в системе лекарственные средства). Связь с управляющей подсистемой осуществляется через глобальный атрибут МЕСТНОЕ_ОБЕЗБОЛИВАНИЕ, имеющий единственное значение «возможно», вычисляющийся в подсистеме с помощью следующего логического правила:

ЕСЛИ АF(ПРОТИВОПОКАЗАНИЯ) ТО [МЕСТНОЕ_ОБЕЗБОЛИВАНИЕ, возможно]

где АF(А) — логическая функция «все значения символического атрибута А ложны». Таким образом, местное обезболивание считается возможным, если нет перечисленных выше противопоказаний.

Значение утверждения [ПРОТИВОПОКАЗАНИЯ.аллергия] выводится по правилу

ЕСЛИ АТ(ВЫЗЫВАЕТ_АЛЛЕРГИЮ) ТО [ПРОТИВОПОКАЗАНИЯ.аллергия]
где АТ(А) — логическая функция «все значения символического атрибута А истинны». Все остальные значения определяются с помощью вопросов к пользователю.

После вычисления коэффициента определенности утверждения о возможности проведения местной анестезии система должна убедиться в том, что его значение не меньше некоторого установленного экспертом предела. Если это так, то система переходит к следующему этапу работы, иначе решение задачи заканчивается.

В результате ввода исходной информации (предложение 2 в сценарии) должны быть получены значения атрибутов ЗАБОЛЕВАНИЕ, ВИД_ВМЕШАТЕЛЬСТВА, ЛОКАЛИЗАЦИЯ, ВОЗРАСТ и др.

Затем в соответствии с локализацией определяются все те нервы, которые могут иннервировать область вмешательства. Для их описания вводится символический атрибут НЕРВ (значения — верхнечелюстной, нижнечелюстной, небный, задние-верхние-альвеолярные и др.). Правила описания иннервации имеют вид

ЕСЛИ [ЛОКАЛИЗАЦИЯ.верхний-1] ∨ [ЛОКАЛИЗАЦИЯ.верхний-2] ТО
[НЕРВ.резцовый] = P-D(1)

где P-D(p) — арифметическая функция, преобразующая вероятность p в соответствующую степень определенности. В правилах подобного вида могут учитываться возрастные особенности, а также возможные отклонения в расположении нервов.

Если цель консультации — анализ ошибок при проведении обезболивания, то система переходит к вводу исходной информации о способе и средствах обезболивания, после чего выводится символический атрибут АНАЛИЗ. Значениями этого атрибута являются указания на возможные причины неудачного обезболивания, например недостаточность способа (блокированы не все нервы, неправильная доза анестетика и др.).

Цель консультации описывается с помощью утверждения «анализ», значение которого определяется по вопросу к пользователю. Если цель консультации — выбор методов и средств обезболивания, система переходит к анализу методов. В ходе этого анализа определяется, какие методы не могут быть применены в данном случае. Метод не применяется, если он не приводит к обезболиванию никакого из найденных нервов, а также если есть прямые противопоказания, например:

ЕСЛИ [ВОСПАЛИТЕЛЬНЫЕ-ПРОЦ-Ч-Л-ОБЛ.есть] ТО [МЕТОД.ин-
фильтрационная]=—100.0

ЕСЛИ ~[НЕРВ.резцовый] ТО [МЕТОД.пров.-У-резц.-отв]=—100.0

Пользователю выводится список допустимых методов с указанием, какой из нервов обезболивается каждым методом (подсистема ВЫБОР-М, предложение 6 сценария). Пользователь должен указать, какие методы он хочет применить, после чего система выведет оценку целесообразности применения выбранного способа обезболивания (предложение 7 сценария).

В первую очередь система проверяет, является ли способ достаточным. Если способ достаточен, то по байесовскому правилу выводится значение утверждения [ПРИМЕНЕНИЕ.целесообразно]. К факторам, определяющим это значение, относятся: количество укулов (числовой атрибут КОЛИЧЕСТВО-УКОЛОВ); общая трудоемкость (утверждение [ПРИМЕНЕНИЕ.трудоемко]; общий риск в случае ошибки ([ПРИМЕНЕНИЕ.имеет-риск]); опасность осложнений ([ПРИМЕНЕНИЕ.опасно-осложнениями]); экономия анестетика с целью снижения его токсического действия ([ПРИМЕНЕНИЕ.экономно]) и др. Подтверждающие и опровергающие веса данного правила отражают точку зрения эксперта на важность перечисленных факторов при принятии решения о выборе способа обезболивания.

Утверждение о трудоемкости способа выводится также по байесовскому правилу, факторами которого являются все возможные методы обезболивания. Опровергающие веса в данном случае равны нулю, подтверждающие выражены с помощью числовых атрибутов, значения которых вычисляются с учетом объективной трудоемкости, а также возрастных и других особенностей пациента. Утверждения об общем риске и опасности осложнений также выводятся с помощью байесовских правил.

Получив оценку целесообразности способа, система сообщает о ней пользователю и предлагает либо изменить решение, либо перейти к выбору анестетика и вазоконстриктора, их дозы и концентрации (предложения 11—14 сценария). При выборе лекарственных средств учитываются объем предполагаемого вмешательства, переносимость пациентом местного анестетика, возраст, токсичность анестетика, его обезболивающее действие и другие факторы. На данном этапе система выводит оценки целесообразности применения каждого из описанных средств, например:

ЕСЛИ [ВЫЗЫВАЕТ-АЛЛЕРГИЮ.тримекаин] ТО [СРЕДСТВО.тримекаин]=—100.0

Определив дозы и концентрации выбранного пользователем анестетика и вазоконстриктора, система задает вопрос: успешно ли проведено обезболивание (вывод значения утверждения «удачно»). Если были допущены ошибки, то система переходит к анализу возможных ошибок (предложение 20 сценария).

Глава 18.

Отечественные экспертные системы на больших ЭВМ

18.1. Система ПЛОТИНА/ЭКСПЕРТИЗА

К. И. Иващенко, Б. С. Кирсанов

18.1.1. Постановка задачи. В народном хозяйстве нашей страны для энергетических и ирригационных целей используется значительное число плотин, дамб и других гидротехнических сооружений. Отклонение от условий нормальной эксплуатации гидротехнического комплекса может привести к аварии. Актуальность этой проблемы обусловлена тем, что аварии крупных гидросооружений могут иметь масштаб национальной катастрофы.. Статистика разруше-

ний за последние 100 лет показывает, что из 15 000 плотин высотой более 15 м, построенных за это время, свыше 100 плотин потерпели серьезные аварии, что повлекло за собой огромный материальный ущерб. Причинами аварий могут быть не только ошибки в проектировании сооружений, но и в нарушении условий нормальной эксплуатации гидроузла. Факторами, усугубляющими серьезность последствий аварии, обычно является малодейственная система аварийного оповещения, отсутствие стратегий действий по уменьшению последствий аварии, а также большая скорость процесса разрушения некоторых плотин (в особенности грунтовых), образования и распространения волны прорыва.

Для обеспечения безопасности эксплуатации гидросооружения необходим постоянный контроль за его работой, регулярная оценка его состояния и анализ безопасности, разработка стратегий действий в случае повреждений и аварий. При оценке состояния эксплуатируемого гидросооружения возникает ряд трудностей, обусловленных следующими причинами.

1. Необходимость анализа очень большого объема исходной информации о гидросооружении. Сюда входят как данные визуальных обследований сооружения и наблюдения по системе контрольно-измерительной аппаратуры, так и данные, получаемые расчетными методами. Состояние сооружения определяется сотнями параметров, поэтому возникают трудности в оперативной обработке информации. В настоящее время при анализе состояния сооружения обычно используется лишь 20—30% доступной информации. Это приводит к снижению объективности получаемых при анализе результатов.

2. Уникальность характера обследуемого гидросооружения: значительные размеры и, как правило, уникальное сочетание природных условий. Критерии, применяемые при оценке состояния гидросооружения, подлежат уточнению в ходе эксплуатации. Поэтому особенности работы конкретного сооружения не могут быть предсказаны достаточно точно при проектировании и уточняются в процессе эксплуатации.

3. Недостаточность и неполнота исходных данных, так как далеко не все процессы, происходящие в сооружении и определяющие его состояние, могут быть охвачены системой инструментальных стационарных измерений из-за пространственной локализации указанных процессов и уникальности и отсутствия разработанной методики измерения.

4. Неточность применяемых расчетных методов, так как не все происходящие в сооружении процессы поддаются расчету из-за неразработанности или исключительной сложности методов и невозможности их сопоставления с натурными данными ввиду отсутствия измеряемых инструментальными средствами параметров.

В настоящее время для оценки состояния гидросооружения применяются визуальные, инструментальные и расчетные методы. Применение каждого метода в отдельности не приводит к удовлетворительному решению задачи. Визуальные методы, субъективные по своей природе, дают неточную информацию. Для успешного применения инструментальных методов требуется развитая сеть контрольно-измерительной аппаратуры, которая имеется не на всех сооружениях. При этом число датчиков в принципе ограничено, а часть из них, в особенности для длительно эксплуатируемых сооружений, выходит из строя. Кроме того, не все явления, происходящие в сооружении, могут быть надежно измерены. При использовании расчетных методов возникает проблема идентификации расчетных моделей, т. е. сопоставления результатов расчетов и наблюдений на предшествующих стадиях эксплуатации и подбор параметров модели, обеспечивающих удовлетворительное совпадение. Естественно, что не все явления, процессы и вызываемые ими состояния сооружения поддаются расчету. Это объясняется по крайней мере двумя причинами: отсутствует (или недоступен исследователю) соответствующий расчетный аппарат; не все явления, определяющие безопасность конкретного обследуемого гидросооружения, известны. Но даже те параметры состояния гидросооружения, которые поддаются расчетной оценке, иногда не с чем сопоставить (это происходит в силу перечисленных выше ограничений инструментальных методов).

Основная проблема состоит в формализации процедуры объединения всех этих методов в единую методику. Однако удовлетворительного решения до сих пор не получено. Это объясняется малой гибкостью методов, невозможностью применять знания, используемые при решении задачи, в их естественной форме, отсутствием эффективных методов, работающих в условиях неполноты и неточности данных. И хотя при оценке состояния гидросооружения используются все три метода, качественное решение задачи получается за счет интеллектуального анализа результатов, выполняемого экспертами. Учитывая, что число экспертов мало, а задача трудоемка и носит массовый характер (в плановом порядке обследуются несколько сотен гидросооружений), только автоматизация ее решения позволяет повысить качество, оперативность и объективность оценки состояния гидросооружений.

18.1.2. Разработка ЭС. Для решения задачи на основе ИС ЭКСПЕРТИЗА (см. гл. 16) был создан действующий прототип ЭС ПЛОТИНА, которая предназначена для определения состояния отдельных элементов гидросооружения и всего комплекса, а также для разработки методики профилактических и ремонтно-восстановительных работ.

В соответствии с технологией проектирования ЭС (см. гл. 13) была сформирована база знаний (БЗ) системы и произведена настройка параметров работы системы ЭКСПЕРТИЗА. Формирование БЗ сопровождалось работой по систематизации методов, используемых при решении задачи, и накопленного экспертами опыта. В ходе совместной с экспертом работы были выделены: основные понятия, позволяющие адекватно описывать и анализировать состояние гидросооружения; зависимости этих понятий и условия применения имеющихся методов.

Все сооружение разбивается на три элемента: плотина, основание, водохранилище. Выделяется три группы понятий:

структурные элементы гидросооружения (плотина, основание плотины, тело плотины, примыкания, водосбросы и т. д.),

источники (способы получения) информации о гидросооружении в целом и его элементах, а также о внешних воздействиях на гидросооружение и видах предельных состояний (сейсмическое воздействие, оползни, напряженно-деформированное состояние элементов сооружения и способы, используемые при оценке состояния элементов сооружения (инструментальный, визуальных наблюдений, расчетный);

виды и причины, количественные и качественные показатели неисправностей, устанавливаемые при непосредственном обследовании гидросооружения (или на основе инструментальных измерений) и являющиеся обычно входными данными (объем оползня, ширина трещины, затухание колебаний, уровень водохранилища и т. д.).

В целях упрощения формулировок правил и процедуры решения задачи для всех трех введенных типов понятий и системы плотина — основание — водохранилище принят обобщенный показатель «степень повреждения». Степень повреждения — это определяемая на основании анализа данного понятия (или вычисляемая в соответствии с правилами) степень отклонения от условий, обеспечивающих максимальную безопасность гидросооружения в целом или его элемента. Введены следующие качественные оценки степени повреждения: незначительное, слабое, среднее, сильное и разрушительное. Кроме обобщенного показателя для представления количественных данных, полученных в результате инструментальных измерений (или расчетов), используется безразмерная характеристика «индекс повреждения». Степень повреждения имеет следующий диапазон изменения по количественной шкале: 0—1,25 (повреждение незначительное), 1,25—3,75 (слабое), 3,75—6,25 (среднее), 6,25—8,75 (сильное), 8,75—10 (разрушительное). Вид функций принадлежности для каждого интервала был определен эвристически, на основе опыта эксперта и анализа статистических данных по авариям крупных плотин.

Использован иерархический принцип взаимодействия понятий системы — в процессе решения задачи по характеристикам понятий низких уровней, соответ-

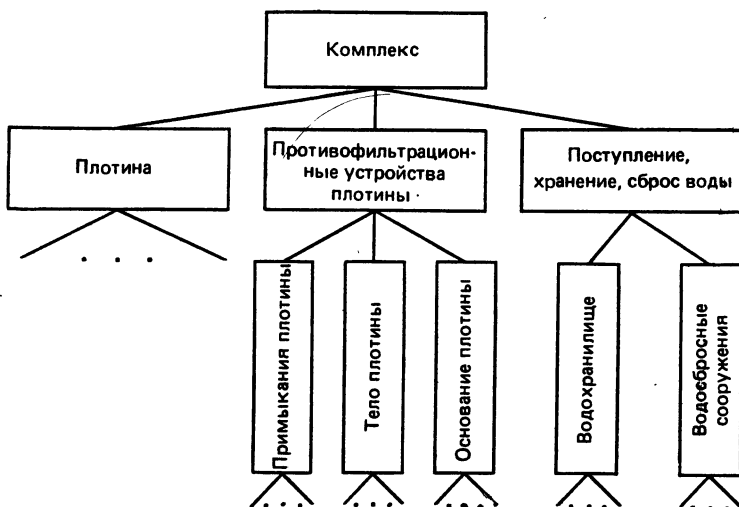


Рис. 18.1. Верхние уровни иерархии понятий системы плотина — основание — водохранилище

ствующих исходным данным, последовательно вычисляются характеристики понятий более высоких уровней. Отнесение понятия к тому или иному уровню, как и построение иерархической системы в целом и ее отдельных цепочек, выполнено экспертом и изменялось в ходе отладки системы. При построении иерархии на каждом уровне эксперт расположил понятия так, чтобы они вносили примерно одинаковый вклад в определение значения степени повреждения на следующем более высоком уровне. Три первых уровня иерархии представлены на рис. 18.1.

Понятия одного уровня, расположенные в иерархической системе слева, имеют приоритет — первоначальные оценки делаются на основе их характеристик. Например, если «степень повреждения» понятия имеет значение «разрушительное», то характеристики понятий, расположенных правее, не учитываются. В вычислениях не участвуют понятия, характеристика которых неизвестна.

Планирование и проведение специальных мероприятий по обеспечению необходимого уровня безопасности гидросооружения только на основе анализа отдельных контролируемых параметров затруднительно (особенно тогда, когда ни один из параметров не достиг предельно допустимого уровня). Поэтому применяется следующая процедура подсчета обобщенного показателя:

$$I = \sum_{i=1}^n \alpha_i I_i / \sum_{i=1}^n \alpha_i \quad (18.1)$$

где I_i — индекс повреждения i -го элемента системы; α_i — коэффициент, определяющий влияние i -го элемента на безопасность системы в целом.

Обобщенный показатель степени повреждения всего гидросооружения вычисляется следующим образом. Для каждого вида предельных состояний и применяемых методов получения информации о состоянии системы (визуального, инструментального и расчетного) по оценкам степени повреждения, установленным в результате анализа контролируемых параметров, вычисляются (на основе формул вида (18.1)) индексы повреждения для элементов системы. Затем по формуле (18.1) определяется индекс повреждения для системы в целом.

Одновременно получается и обобщенный показатель — степень повреждения. В соответствии со значениями индекса повреждения и степени повреждения назначается перечень предупредительных или ремонтно-восстановительных мероприятий по обеспечению безопасности системы плотина — основание — водохранилище, а также отдельных ее элементов.

18.1.3. База знаний. Все знания, используемые системой при решении задачи, можно разделить на следующие группы:

- 1) понятия предметной области и иерархия этих понятий;
- 2) таблицы значений параметров, соответствующих внешним воздействиям (землетрясение, оползни, паводки);
- 3) специализированные пакеты прикладных программ, реализующие расчетные методы;
- 4) справочная информация по различным структурным элементам комплекса;
- 5) статистическая информация по данному сооружению за период наблюдений и информация по аналогичным сооружениям (в СССР и за рубежом);
- 6) формализованный в виде продукционных правил опыт экспертов по оценке состояния сооружений;
- 7) знания, необходимые для управления работой системы: разделение на подзадачи, порядок их разрешения, стратегии работы решателя для каждой подзадачи;
- 8) знания, отражающие связи понятий и предназначенные для использования при объяснении работы системы.

Понятия предметной области представлены с помощью ИС ЭКСПЕРТИЗА схемами объектов. Специальных средств для представления иерархии в ИС ЭКСПЕРТИЗА нет, и для этой цели были использованы метаправила. Табличная информация хранится в базе данных (БД) ЭС, и обращение к ней осуществляется стандартными средствами (функция ИСКАТЬ). Статистические и справочные знания хранятся в специально организованной БД, обращения к которой оформлены в виде внешних процедур ЭС. В виде внешних процедур оформлены и все программы для расчетных методов. Среди продукционных правил выделены эмпирические и расчетные. Эмпирические отражают опыт эксперта и обрабатывают информацию качественного характера — результаты визуальных наблюдений и оценки состояния различных элементов гидросооружения. Расчетные правила описывают условия применения расчетных методов. Действием этих правил обычно является обращение к внешним процедурам или к табличной информации.

Управляющая информация реализуется с помощью объектов управляющей памяти и метаправил. Информация для объяснений хранится в описаниях схем объектов. При реализации системы использовалась выявленная иерархия объектов. Наличие иерархии объектов позволило: убрать из правил описание связей различных понятий предметной области — в правилах остались только условия их применимости; эффективно управлять процессом получения решения — осуществлять структуризацию рабочей памяти и планирование порядка применения правил; повысить наглядность работы системы при объяснении. В настоящее время БЗ системы включает ~200 описаний объектов и ~500 правил, что позволило адекватно описывать состояние реального гидротехнического сооружения (Чарвакского гидроузла). Система установлена на ЕС-1061, время отклика не превышает 10 с, время решения — 10 мин.

В ходе создания ЭС была существенно доработана методика определения состояния гидросооружения, что позволило повысить объективность и оперативность обследования гидросооружений. Особенно перспективно использование ЭС для гидросооружений, основная часть показателей состояния которых имеет качественный характер. Такая ситуация характерна для сооружений с длительным сроком эксплуатации. Подобных сооружений много, и их число, естественно, с каждым годом увеличивается. Ведутся работы по дальнейшему совершенствованию системы и переводу ее на ПЭВМ.

18.2. Промышленная система ДАМП/ЭКСПЕРТИЗА

М. М. Виньков, Б. Е. Демин, Б. С. Игнатов, В. Н. Игнатьев

18.2.1. Постановка задачи. Из всех видов деятельности при разработке программного обеспечения наибольших интеллектуальных затрат требует отладка программ. Она обычно выполняется при громадном организационном и внутреннем напряжении, которое вызвано стремлением установить ошибку как можно быстрее. Несмотря на наличие разнообразных методов отладки программного обеспечения, полностью формализовать этот процесс не представляется возможным. Отладка программ — вообще трудоемкий процесс, а для систем телеобработки он еще более усложняется.

В системах телеобработки на основе телемонитора СИРЕНА это обусловлено следующим. Во-первых, отсутствует защита памяти: управляющие структуры и программы, прикладные модули, рабочие области не гарантированы от возможных разрушений чаще всего из-за неправильно работающих прикладных программ. Это приводит к тому, что неверно запрограммированная задача может вызвать аварийное завершение других задач (в том числе отлаженных), а также аварийную ситуацию всей системы. Во-вторых, диагностировать аварийную ситуацию приходится обычно тогда, когда области памяти, необходимые для анализа, уже освобождены телемонитором и используются другими задачами. В результате ошибок в прикладных программах, приводящих к нарушению управляющих структур телемонитора, возникают тяжелые аварийные ситуации, которые в дальнейшем будут рассматриваться. Частота их возникновения может достигать нескольких раз в неделю или даже в день и зависит от степени отлаженности прикладных программ.

Анализ аварийных ситуаций требует большой работы по разбору дампов и обнаружению ошибок в прикладных программах. Объем дампа, как правило, составляет 400—600 страниц текста в шестнадцатеричном представлении. Лишь немногим системным программистам удалось (после нескольких лет практики) достичь высокой результативности в анализе дампов аварийных ситуаций (свыше 80% случаев). Деятельность в этой области требует от человека глубоких знаний о механизмах управления, реализованных в телемониторе, типовых ошибках в прикладных программах, наиболее часто приводящих к аварийным ситуациям, и о том, как эти ошибки проявляются на дампе.

Острая необходимость в высококвалифицированных специалистах по анализу аварийных ситуаций привела к созданию ЭС ДАМП, которая позволила решать задачу анализа аварийных ситуаций специалисту средней квалификации. В каждом конкретном случае, разбирая дамп, сформированный при аварийном завершении системы телеобработки, ЭС ДАМП либо указывает программы, ошибки в которых привели к возникновению аварийной ситуации, либо рекомендует, какой режим работы телемонитора следует установить, чтобы при повторном возникновении аварийной ситуации ее анализ дал значимый результат.

18.2.2. Разработка ЭС. Наблюдения за работой экспертов при анализе аварийных ситуаций показали, что наряду с решением сложных интеллектуальных задач экспертам приходится выполнять большой объем рутинной работы. Например, обнаружение адреса нарушения цепочки памяти, требующее выполнения многочисленных арифметических действий над шестнадцатеричными числами. Поэтому в работе экспертов, проводящих анализ аварийных ситуаций, естественно выделить два уровня:

интеллектуальный: оперирование сложными понятиями, сформулированными экспертами на основании их предшествующего опыта;

рутинный: извлечение информации из дампа аварийной ситуации таким образом, чтобы используемые на первом уровне понятия приобрели содержательные значения применительно к анализируемой ситуации.

Изучение работы экспертов привело к принятию следующей гипотезы (рис. 18.2). На интеллектуальном уровне производится классификация анализируемых ситуаций (как известно, построение классификационных систем лежит в основе

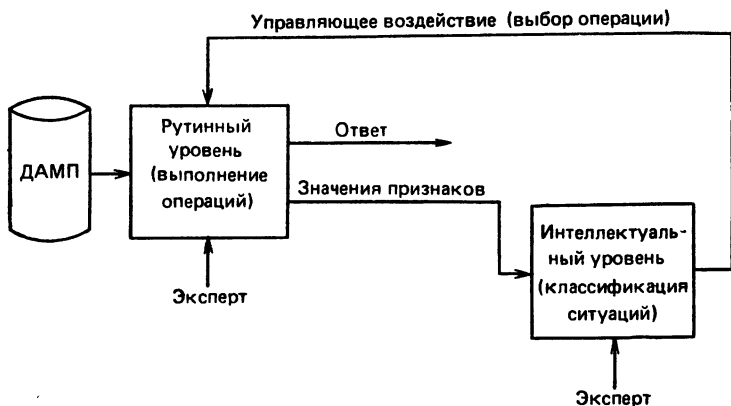


Рис. 18.2. Функциональная схема процесса анализа экспертом дампа АС

многих видов интеллектуальной деятельности). Чтобы решить поставленную задачу, эксперту необходимо отнести анализируемую ситуацию к одному из известных ему классов. Значения признаков, используемых для описания ситуаций, могут быть получены либо непосредственно из дампа аварийной ситуации (с помощью выполнения некоторой рутинной работы), либо с помощью логического вывода на основе знания о связях между значениями признаков.

На рутинном уровне выполняется совокупность операций, носящих регулярный характер. Эксперт выполняет ту или иную операцию, когда возникает потребность в получении соответствующей информации. Некоторые операции ставятся в соответствие классам ситуаций и используются для получения ответа на основной вопрос, решаемый при анализе аварийной ситуации, — идентификация (установление имени или списка имен) программ, являющихся вероятными «виновниками» аварии.

В результате выполнения экспертом рутинных операций получается исходная информация, анализируется им, после чего по результатам анализа эксперт определяет, какую информацию необходимо получить для дальнейшего анализа и, соответственно, какую выполнить рутинную операцию. Когда задача классификации данной ситуации решена, выполняется операция, соответствующая тому классу, к которому экспертом отнесена анализируемая ситуация. Выполнение этой операции дает окончательный ответ на поставленный вопрос.

При анализе аварийных ситуаций была выявлена и построена иерархическая классификация аварийных ситуаций. При описании классов ситуаций использованы как признаки, характеризующие сущность нарушения управляющих структур, так и признаки, характеризующие «сценарий» того, как эти нарушения произошли (пример сценария: адрес в запросе указывает на границу раздела динамической памяти).

При реализации ЭС ДАМП было выбрано ИС ЭКСПЕРТИЗА, хорошо подходящее для решения задач классификации сложных структурированных объектов. Для использовании ИС ЭКСПЕРТИЗА для реализации ЭС ДАМП объектом разработки стали база знаний и присоединенные процедуры (рис. 18.3). При решении задачи анализа аварийных ситуаций хранимые в системе знания обрабатываются интерпретатором знаний ИС ЭКСПЕРТИЗА. Конкретные факты анализируемой ситуации, представленные в виде экземпляров объектов с означенными атрибутами, хранятся в рабочей памяти. Инициирование решения задачи, выдача окончательных и промежуточных результатов, а также получение необходимой информации от пользователя осуществляются с помощью диалогового компонента.

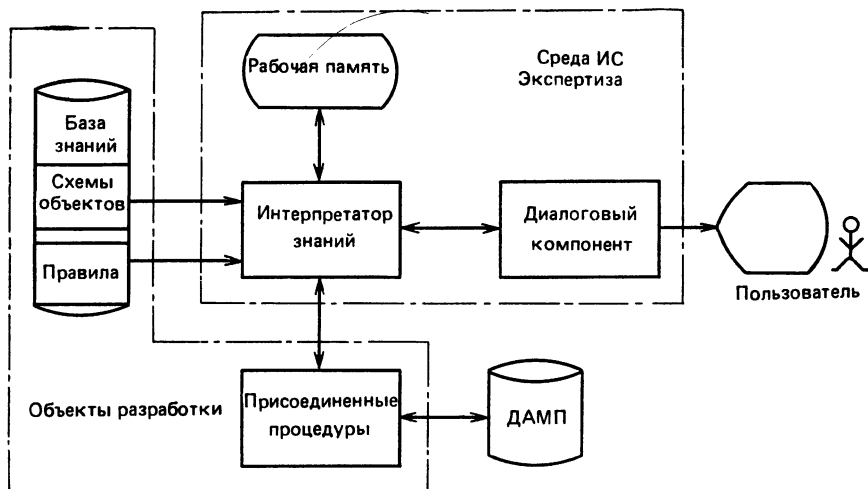


Рис. 18.3. Схема построения ЭС ДАМП/ЭКСПЕРТИЗА

18.2.3. База знаний. Основным компонентом ЭС ДАМП является база знаний (БЗ), где в виде схем объектов и правил представлены как теоретически обоснованные знания о механизмах управления телемонитора СИРЕНА, так и основанные на опыте экспертов знания о возможных ошибках и их проявлениях, целесообразной последовательности действий при анализе и т. д. Схемы объектов описывают три вида сущностей: управляющие структуры телемонитора СИРЕНА, процессы и события. Правила устанавливают связи между объектами, объединяются в наборы в соответствии с контекстным типом (например, тип «проверка целостности цепочки терминальной памяти»). Специальные метаправила содержат сведения о том, какие наборы правил и в какой последовательности должны проверяться. Метаправила задают контекстный граф, который определяет отношения между наборами правил и управляет интерпретатором знаний во время работы ЭС ДАМП, отражая процесс рассуждения экспертов. Таким образом, в БЗ представлены понятия, используемые экспертом на интеллектуальном уровне. На рис. 18.4 приведен фрагмент контекстного графа

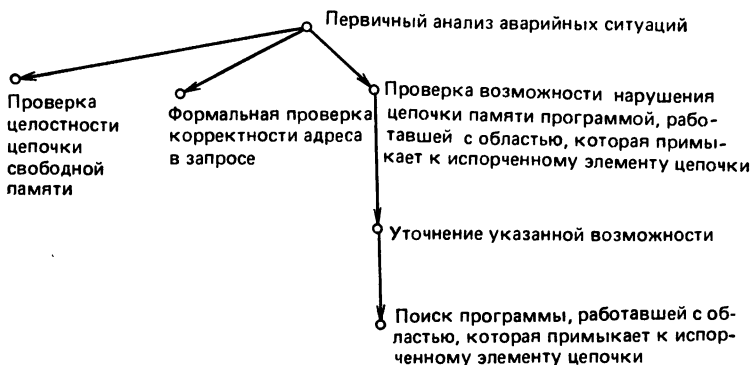


Рис. 18.4. Фрагмент контекстного графа

фа (набором правил соответствуют вершины графа, метаправилам — дуги), задающий последовательность наборов правил, в которой интерпретатор обрабатывает аварийную ситуацию, отнесенную ЭС к классу «элемент цепочки свободной памяти нарушен программой, работавшей с соседней областью памяти».

Для выполнения анализа аварийной ситуации требуется информация, получаемая на рутинном уровне. С этой целью в ЭС введены присоединенные процедуры. Интерфейс с этими процедурами осуществляется с помощью правил. Результат выполнения процедур можно рассматривать как сложные действия правил. В отличие от знаний, используемых на интеллектуальном уровне, знания, используемые на рутинном уровне, скрыты в присоединенных процедурах и не представлены в системе явно.

ЭС ДАМП содержит в БЗ порядка 400 правил и находится в эксплуатации в ряде организаций. Время анализа одной аварийной ситуации составляет около 3—5 мин при работе на ЭВМ ЕС-1066 (на анализ дампа средней сложности у специалиста по телемонитору СИРЕНА уходит 6—8 ч). Дальнейшее развитие системы связано с повышением достоверности результатов ее работы (за счет ввода в систему новых знаний), а также с повышением информативности даваемых системой ответов.

Хотя задача анализа аварийной ситуации в своей первоначальной постановке отличалась от типовых задач классификации, так как перечень возможных решений заранее неизвестен, оказалось, что она может быть сведена к задаче классификации. Такой подход может быть распространен и на решение аналогичных задач, связанных с другими телемониторами, отличными от телемониторов СИРЕНА и КАМА.

18.3. Система СОНЭТ/ЭКСПЕРТИЗА

Б. С. Кирсанов, С. В. Маслов

18.3.1. Постановка задачи. Развитие электронной аппаратуры, в том числе вычислительной техники, немислимо без параллельного развития методов ее контроля и поиска технических неисправностей. Контроль и поиск неисправностей, осуществляемый вручную экспертами, не всегда эффективен, так как это довольно трудоемкая и длительная процедура. Эксперты помимо интеллектуальной работы по локализации неисправностей вынуждены выполнять и тяжелые рутинные операции по вычислению сигналов внутри схемы, работу со справочниками и т. д. Кроме того, число таких специалистов весьма ограничено. Этими и другими причинами и обусловлено появление различных автоматизированных средств контроля и поиска неисправностей, встроенных и внешних. Использование встроенных средств часто приводит к недопустимому увеличению стоимости устройства, габаритов и т. д. В этих случаях используют внешние средства контроля.

Алгоритм решения задачи при внешнем контроле строится на основе формальных моделей и представляет собой жесткую, заранее заданную последовательность действий и вычислений, приемлемую с точки зрения имеющихся ресурсов.

Использование формальной модели часто полностью и эффективно решает задачу, например в тех случаях, когда для схемы заранее известны возможные неисправности и их проявления, а также соответствия между неправильным поведением схемы на выходе и неисправностью. Однако, как показал анализ, большая часть схем не укладывается в рамки этого ограничения. В этом случае решить задачу можно, используя в ЭВМ знания эксперта не только об общих и конкретных законах функционирования цифровых схем и общих методах поиска неисправности, но и интуитивные неформальные знания по решению задачи.

Использование методов инженерии знаний для решения задачи поиска технических неисправностей оправдано тем, что число высококвалифицированных экспертов ограничено, а задача носит массовый характер и часто решается на

практике. Применение методов инженерии знаний к решению указанной задачи не только оправдано, но и возможно, так как:

эксперты в данной области есть и они в состоянии описать свои знания на естественном языке;

поиск неисправности занимает у эксперта несколько часов и не является чрезмерной сложной или чрезмерно простой задачей;

правильность работы системы легко проверить на реальных задачах и тестовых примерах;

задача не использует в значительной степени широкого спектра общих сведений о мире.

Иными словами, можно утверждать, что применение ЭС для решения задачи поиска неисправностей оправдано, возможно и соответствует методике инженерии знаний.

Справедливость этого утверждения доказывается появлением и высокой эффективностью работы ряда коммерческих ЭС, таких, как DART, IN-ATE, и др. (см. гл. 14). Примером решения проблемы может служить рассматриваемая ниже гибридная система для Обнаружения и локализации Неисправностей в цифровых схемах Электронной Техники (СОНЭТ).

18.3.2. Функционирование ЭС. Работу системы можно разделить на два основных этапа

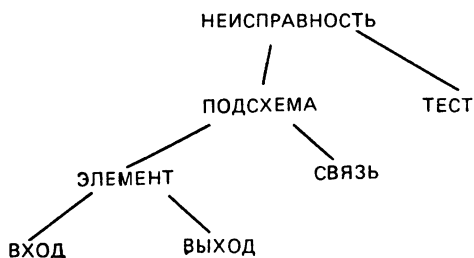
1. Этап контроля. На этом этапе проверку работоспособности объекта контроля (цифровой схемы) осуществляет аппаратно-реализованная модульная система контроля (МСК). МСК подает тестовые наборы на входы объекта контроля (ОК) и анализирует реакции на выходе. Последовательность подаваемых тестовых наборов одинакова для всех классов ОК, и знание о соответствии между конкретной последовательностью тестовых наборов и какой-либо группой тестируемых элементов отсутствует (вся последовательность сформирована таким образом, что в среднем для схем всех классов она может выявить около 80 % неисправностей). В случае несовпадения ожидаемых и фактических реакций на выходах ОК из МСК в ЭС поступает информация о виде несовпадения.

2. Этап локализации неисправности. На этом этапе с помощью методов, заложенных в БЗ, осуществляется анализ работоспособности компонентов схемы и выявляется точка возникновения неисправности. Под точкой возникновения неисправности понимается одиночная неисправность одного из трех типов: неисправный выход элемента (постоянный непредусмотренный уровень сигнала, возникающий на одном из выходов элемента при любом входном воздействии); неисправный вход элемента (непредусмотренный постоянный уровень сигнала, возникающий на одном из входов); неисправная единичная связь между элементами (в простейшем случае это участок электрической цепи между выходом одного элемента и входом другого, не содержащий разветвлений). Подавляющее большинство неисправностей, возникающих на практике, относится к одному из трех указанных типов.

На этапе локализации неисправности система (как и человек-эксперт) сталкивается с необходимостью решения ряда четко формализованных подзадач. Часть из них представляется в виде знаний, хранящихся в БЗ системы, часть — на основе взаимодействия решателя с аппаратно-реализуемыми средствами (например, МСК) или программно-реализуемыми средствами (различными пакетами прикладных программ). Управление выбором подзадач осуществляется на основе знаний эксперта, заложенных в ЭС. К наиболее трудоемким подзадачам относятся:

Начальная загрузка. Цель этой подзадачи — выделение подсхемы общей схемы, содержащей неисправность (в структуру подсхемы включаются элементы и связи, потенциально способные влиять на все неправильные выходы схемы, — назовем их кандидатами на неисправность — плюс элементы и связи, информация о которых может помочь при анализе работоспособности кандидатов на неисправность), и формирование на этой основе элементов рабочей памяти ЭС. Информация, передаваемая в рабочую память, включает: структуру подсхемы, содержащую неисправность; ожидаемые и известные реальные значения сигнала

Рис. 18.5. Иерархическая структура объектов рабочей памяти



лов на входах и выходах элементов схемы; общую характеристику подсхемы.

Моделирование. Блок моделирования (БК) позволяет моделировать работу на заданной последовательности тестовых наборов, одного или нескольких связанных элементов или всей подсхемы, содержащей заданную неисправность. Этот блок основан на представлении структуры ОК в виде модели альтернативных графов [Убар, 1980], одной из характеристик которой является возможность представления как комбинационных, так и последовательных (т. е. обладающих памятью) элементов цифровых схем.

Диалог с оператором. Автоматический съем реальных уровней сигналов МСК осуществляет лишь с выходных контактов ОК. Когда этого недостаточно, решатель может потребовать от человека непосредственного «прощупывания» внутренних точек схемы (возможные точки «прощупывания» известны решателю из структуры схемы) и некоторых других действий. Одной из целей ЭС является сокращение указанных точек «прощупывания».

Система СОНЭТ реализована на основе ИС ЭКСПЕРТИЗА с помощью «настройки» на данную проблемную область. Настройка осуществляется заполнением БЗ ИС ЭКСПЕРТИЗЫ (рабочей памяти и продукционной памяти) и заданием соответствующих параметров режима работы решателя.

18.3.3. База знаний. Определение структуры БЗ осуществлялось совместно со специалистами в области цифровых схем. В ходе этой работы были выяснены основные понятия данной предметной области — декларативные знания, наличие которых является достаточным для описания любой рабочей схемы и любой рабочей ситуации в ходе решения задачи. В рабочей памяти декларативные знания представляются в виде объектов. В структуру объекта входит имя объекта, список его атрибутов характеристик с соответствующими значениями и различная вспомогательная информация (время создания значений атрибутов, источник создания, коэффициент достоверности и т. д.). Взаимосвязь объектов можно представить в виде иерархии (рис. 18.5).

Здесь объект НЕИСПРАВНОСТЬ служит для отражения общих характеристик проявления неисправности в ОК, объект ПОДСХЕМА — для отражения общих характеристик подсхемы, содержащей неисправность. Объект ТЕСТ отражает свойства инициального тестового набора — набора (последовательности наборов), при подаче которого схема устанавливается в определенное начальное состояние, которое не зависит от входных воздействий в прошлом. (Состояние схемы определяется выходными параметрами содержащихся в ней последовательных элементов.) Моделирование работы подсхемы в дальнейшем будет осуществляться с инициального тестового набора. Объект СВЯЗЬ служит для отражения характеристик связей между элементами подсхемы, объект ЭЛЕМЕНТ — характеристик элементов подсхемы, объект ВХОД — характеристик входов элементов, объект ВЫХОД — характеристик выхода элементов. Объекты НЕИСПРАВНОСТЬ, ПОДСХЕМА, ТЕСТ уникальны в рабочей памяти. Число экземпляров объектов СВЯЗЬ, ЭЛЕМЕНТ, ВХОД, ВЫХОД соответствует числу аналогичных сущностей в рассматриваемой подсхеме.

Знания о преобразовании объектов (процедурные знания) представляются в виде правил продукционной памяти. Эти знания можно классифицировать как

формальные знания общих законов функционирования цифровых схем и методов поиска неисправности и неформальные знания, касающиеся конкретных классов схем, приобретенные экспертом на основе интуиции и опыта работы.

В основные шаги поиска решения, основанные на формальных методах, входят:

начальная загрузка, моделирование и диалог с оператором, рассмотренные выше;

выделение элементов и связей, потенциально способных влиять одновременно на все неисправные выходы;

проверка работоспособности элементов, часть входов и выходов которых совпадает с входными и выходными полюсами ОК;

анализ работоспособности элементов на основе их поведения на предыдущих тестовых наборах;

проверка работоспособности элемента на основе значений сигналов на входах и выходах;

нахождение инициального тестового набора, устанавливающего схему в начальное состояние.

К неформальным шагам поиска решения, используемым в системе СОНЭТ, относятся:

определение наиболее «уязвимых» с точки зрения работоспособности групп элементов ОК;

определение кандидатов на неисправность на основе знаний о соответствии между видом проявления неисправности в схеме и точкой возникновения неисправности, когда это соответствие не описывается четко формализованной причинной зависимостью, но проявляется в большинстве случаев;

определение кандидатов на неисправность в ситуациях, для которых у эксперта имеется уже готовое решение.

Все эти формальные и неформальные шаги используются в двух основных классах правил (знаний). Первый не требует от человека-оператора непосредственного вмешательства по съему сигналов внутри схемы.

Если знаний первого класса оказывается недостаточно для локализации неисправности, решателю становится доступен второй класс правил, содержащий правила с действиями, требующими ручного вмешательства.

В качестве примера приведены три простых типовых правила, используемые в системе СОНЭТ.

Правило 1 является типичным примером рассуждения эксперта: если на прямых и инверсных выходах элементов наблюдаются одинаковые уровни сигналов, то один из данных выходов неисправен. Коэффициент определенности правила велик, но оно недостаточно, так как причинной ситуации может быть также непредусмотренная «обратная» передача сигнала на выход рассматриваемого элемента через связь с неисправным входом последующего элемента. Правило использует информацию, содержащуюся в атрибутах «РЕАЛЬНОЕ — ЗНАЧЕНИЕ — СИГНАЛА» из двух экземпляров объекта ВЫХОД, соответствующих прямому и инверсному выходам элемента.

Правило 2 — пример обращения к программно-реализованной подзадаче: если в ходе предыдущей работы определить исправность компонента (входа, выхода, связи) не удалось, сделай заключение о его работоспособности на основе моделирования неисправности его работы и сравнения результатов с показаниями реального ОК. Правило имеет низкий приоритет, так как обращение к блоку моделирования требует относительно больших временных затрат и используется после применения других способов. Правило имеет в правой части функцию обращения к БМ, входными параметрами которого являются вид моделируемой неисправности и адрес моделируемого компонента, а выходными параметрами значения ДА, если неисправность совпадает с показаниями реального ОК, и НЕТ в противном случае. Правило недостаточно, так как моделирование неисправности компонента не гарантирует единственности этого кандидата на неисправность.

Правило 3 является примером метаправила и управляет переходом в другой подкласс правил: если требуется проанализировать работу элемента типа RS-триггер, перейди в подкласс правил RS-класс. В правой части правила — функция «ПЕРЕЙТИ_В_ПОДКЛАСС» с параметром RS-класс. Коэффициент определенности правила 1. Правило использует информацию, содержащуюся в атрибуте «Тип» экземпляра объекта ЭЛЕМЕНТ.

Для оптимизации работы с рассмотренными выше знаниями в ЭС СОНЭТ заданы следующие параметры режима работы решателя:

- тип вывода от цепи;
- вывод множественный (т. е. использующий коэффициент определенности и систему альтернативных миров);
- возможность получения нескольких решений;
- запрет создания новых экземпляров объектов в ходе работы (все экземпляры создаются при начальной загрузке) и на этой основе запрет анализа путей решения, требующих этого создания.

В настоящее время система СОНЭТ находится на стадии исследовательского прототипа, способна локализовать одиночные ошибки на входах и выходах элементов схемы, а также ошибки связей между элементами. База знаний системы содержит 150 правил и описание семи схем объектов. Объекты рабочей памяти в зависимости от сложности анализируемой схемы могут иметь до 100—120 экземпляров. На локализацию неисправности реальной схемы затрачивается до 12—15 мин.

Переход к промышленному образцу ЭС будет связан с аппаратной реализацией ряда компонентов ЭС, блока моделирования, а также интерфейсов между вычислительной системой и аппаратно-реализуемыми компонентами.

18.4. Система ЛЕДИ-Z

З. В. Рахманова

18.4.1. Постановка задачи. Одной из задач медицинской кибернетики является создание систем, обладающих способностью к накоплению и обобщению знаний медиков в различных областях медицины, к принятию решений и объяснению рекомендуемых решений. Известным примером таких систем является описываемая в данном разделе система ЛЕДИ-Z, осуществляющая представление медицинских знаний о реанимационных болезнях как болезнях всего оживленного организма. Такая система является как бы микромоделью медицины в целом, так как требует одновременно знаний врачей многих специальностей, определяющих деятельность врача в тех или иных ситуациях для достижения глобальной цели — здоровья пациента. Эта система должна уметь определять состояние различных физиологических систем организма, при необходимости дифференцировать их, вырабатывать и проверять гипотезы возникновения патологических состояний, прогнозировать течение заболевания и, как частный случай, исход его; определять изменение состояний в динамике; рекомендовать вид и схему лечения с учетом показаний или противопоказаний и других факторов (возврат, тяжесть состояния, степень поражения различных физиологических систем, взаимосвязанность патологических состояний, сочетаемость и кумулятивность лекарственных средств и т. д.), экстраполировать состояние больного с учетом действия лечения; корректировать лечение.

Как видно, перечень этих задач подобен тем, которые нужно решать человеку в различных системах управления: определение состояния объекта, выявление рассогласований, выявление и устранение их причин, прогноз эволюции состояния объекта и т. п.

При построении системы принимались в расчет и следующие соображения: не все цели управления могут быть выражены в виде количественных соотношений и между рядом параметров, оказывающих влияние на процесс управления, не удастся установить точных, выявленных эмпирическим путем, количественных

соотношений; процесс достижения цели многошаговый и содержание каждого шага не может быть однозначно определено; объект эволюционирует во времени, меняются его структура и функции, что приводит к эволюции самого процесса управления; элементы, входящие в структуру управляемого объекта, имеют активную природу и их поведение может противоречить целям управления; объекты имеют иерархическую структуру, для каждого уровня может быть свое решение; элементы системы объединены большим количеством различных связей.

18.4.2. Функционирование ЭС. Основой разработки системы ЛЕДИ-Z стал модифицированный набор блоков общей схемы решения задач, ставшей уже классической в ситуационном управлении сложными техническими и организационными системами; 1) входной преобразователь, функцией которого является получение сведений об объекте и среде; 2) анализатор — определение конфликтности ситуации и необходимости вмешательства системы управления в процесс, протекающий в объекте управления; 3) интерпретатор — пополнение модели знаний, преобразование текущей ситуации в полную; 4) адаптатор — переоценка ситуаций; 5) классификатор — обобщение и конкретизация, определение принадлежности конфликтной ситуации к определенному классу; 6) коррелятор — знание законов функционирования объектов, ограничения на управляющие воздействия; 7) блок обучения — формирование иерархического множества обобщенных ситуаций, построение системы правил вывода; 8) экстраполятор дедуктивный, индуктивный и абдуктивный выводы, оценивание альтернативных решений, «проигрывание» решений; 9) накопитель — хранение и использование знаний о рассмотренных объектах управления и выведенных системой решений по управлению ими; 10) выходной преобразователь — выдача решений.

Конкретизируем назначение каждого из вышеперечисленных блоков для реализации функций-прототипов системы ЛЕДИ-Z. Рассмотрим схему функционирования лечебно-диагностической системы ЛЕДИ-Z (рис. 18.6), где блоки выделены штриховыми линиями и обозначены цифрами от 1 до 9.

Блок 1 осуществляет функции входного преобразователя. С карты обследования, где исследуемые медицинские параметры расположены по порядку, в каком их наблюдают врачи при осмотре больного, конкретные значения медицинских признаков заносятся в формализованную историю болезни (ИБ). В блоке 2 совмещены функции анализатора и интерпретатора. Здесь содержатся сведения о медицинских признаках, в частности об их принадлежности к определенному классу, и диапазонах, характеризующих нормальное и патологические состояния. При вхождении значения признака в «патологический» диапазон запускается механизм блока 3. Дополненная новой информацией ситуация записывается в машинную историю болезни (МИБ, блок 8), которая накапливает как данные из формализованной ИБ, так и всю информацию, порождаемую системой из знаний об этих данных (из всех блоков) и используемую этими блоками для решения различных задач в лечебно-диагностическом процессе. Блок 3 осуществляет диагностику состояний, активизируя для этого диагностические правила. Диагностические правила включают признаки или их сочетания (иногда даже противоречивые), необходимые для того, чтобы диагноз мог быть поставлен в различных ситуациях. Сюда входят также те данные, которые не несут основной информации, но способны повысить правдоподобность диагноза. В диагностических правилах последовательность ситуативных структур, отражающих клинику, биохимию и патофизиологию заболевания, представляется в виде И — ИЛИ графов. Любое из определяемых системой состояний является узлом дерева болезней. Для построения дерева болезней были выбраны достаточно широкие области терапии, как, например, поражения почек, печени, сердечно-сосудистой системы. Внутри каждой области на основе патогенетического механизма выделяются подклассы. Например, подклассами поражения почек являются почечная недостаточность, воспалительный процесс в почках, почечно-каменная болезнь, нефроз, инфаркт почек. Далее эти подклассы подразделяются на более конкретные диагнозы по виду, типу, степени тяжести, фазе, остроте течения и т. д.

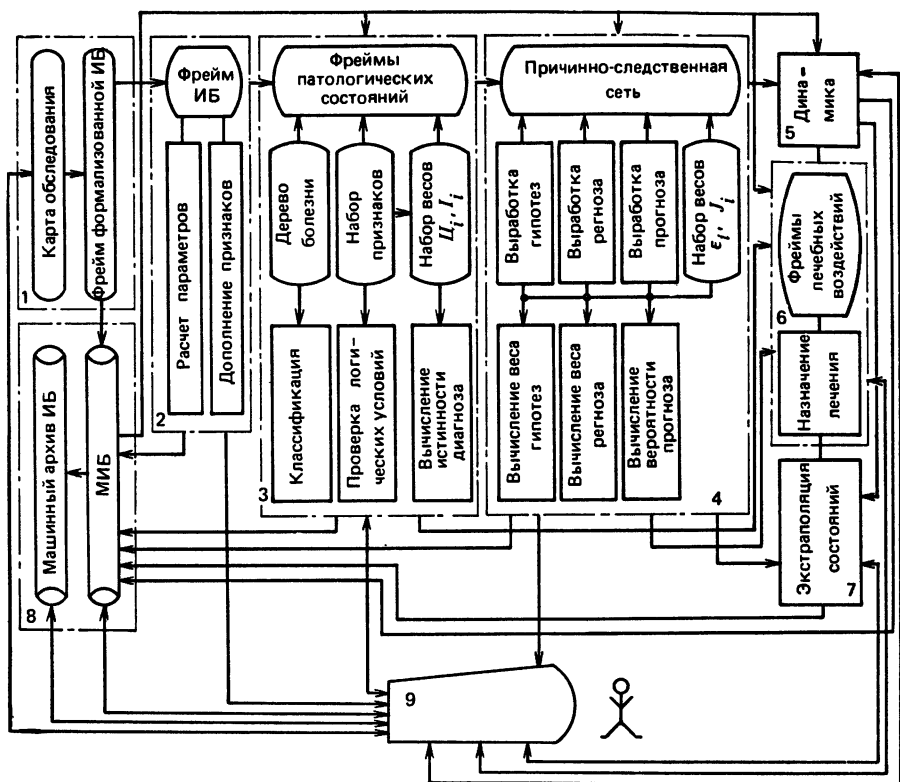


Рис. 18.6. Схема функционирования лечебно-диагностической модели

При построении и использовании деревьев диагнозов возможно три случая использования иерархичности.

Случай 1. При пересечении списков проявлений (характеристик) для всех подузлов каждого узла дерева формируется список общих проявлений. В этом случае при формировании деревьев диагностических правил используется индуктивный вывод, а при эксплуатации для диагноза — дедуктивный вывод. Этот способ удобен и при недостатке информации, когда модель может диагностировать патологию на грубом уровне и направлять внимание врача на более специализированное обследование.

Случай 2. Клиническая картина проявлений поддиагнозов одного и того же диагноза так разнообразна, что пересечение списков этих проявлений ничего не дает, и вывод диагноза как при формировании диагностического дерева, так и при эксплуатации — индуктивный, от дифференцированного к общему.

Случай 3. При пересечении списковых структур получается ряд признаков, в своей совокупности не представляющих особой диагностической ценности. Тогда этот ряд признаков выделяется в формальный симптомокомплекс.

В процессе диагностики используются правдоподобные процедуры вывода. Выводам приписываются качественные оценки, такие как истинность диагностируемого состояния (истинно, почти истинно, неистинно), дифференциация степени тяжести (легкая, средняя, тяжелая и т. п.). Диагностический процесс можно представить как растущую пирамидальную сеть, ее конфигурация может ме-

няться в процессе функционирования. В основе этой смены лежит причинно-следственная связь, и патологический процесс можно представить в виде причинно-следственной сети (см. блоки), узлами которой являются патологические состояния, а дуги определяют временные отношения, характер связей, отношения причина — следствие и веса этих связей. В основе блока 3 лежит довольно сложная структура, совмещающая функции классификатора, коррелятора, адаптера, а также некоторые функции экстраполятора.

Истинность диагноза для первого шага работы модели при выводе определяется по формуле

$$I_1 = \sum_{i=1}^n C_i / C_{\text{абс}}.$$

где C_i — ценность признака для данного диагноза в диапазоне $[0; 1]$; $C_{\text{абс}}$ — абсолютная ценность — сумма ценностей минимального количества признаков с максимальными значениями ценностей, заведомо голосующая за наличие диагностируемой патологии у больного.

Для диагнозов на последующих шагах истинность определяется как

$$I_k = \frac{\sum_{i=1}^n C_i}{C_{\text{абс}}} \sum_{j=1}^p I_{(k-1)j},$$

где $I_{(k-1)j}$ — истинность диагнозов, поставленных на предыдущих шагах работы модели; $j=1, \dots, p$ — число диагнозов, участвующих в определении данного диагноза.

Таким образом, оценка истинности диагноза является динамической, формируемой в процессе работы модели. В случае процесса, усиливающегося рядом дополнительных факторов, возможно $I_k > 1$, в случае недостаточного обследования или неполностью развившегося (или затухающего) процесса $I_k < 1$. Оценка истинности участвует также в вычислении силы транзитивности причинно-следственных связей (иными словами, верности предсказания патологических состояний или вероятности гипотез их возникновения).

Блок 4 помогает врачу выявить причины и следствия заболевания. Двигаясь вверх или вниз по дереву вывода, можно выявить цепочки причинно-следственных связей, основываясь на свойстве транзитивности отношения причина-следствие. Так как для построения гипотезы и построения прогноза должны производиться идентичные вычисления (только направленные в разные стороны от узла, представляющего собой патологическое состояние), удобно для управления процессом вывода в ту или другую сторону пользоваться инвертированными отношениями « x — причина y » и « y — следствие x ». При наличии в запросе слов «прогноз», «гипотеза», «причина», «следствие» они будут ключевыми для организации поиска в системе выработки гипотез или прогноза.

После работы диагностического алгоритма могут сформироваться различные сочетания субдиагнозов. Врачу же нужен супердиагноз — диагноз, выведенный в соответствии с деревом болезни и представляющий собой полное описание какой-либо физиологической системы, например поражения почек. Используя значение причинно-следственных связей и имея подтверждение патологическим состоянием из списка причинно-следственных связей, модель способна связать имеющиеся супердиагнозы в единую систему для всего организма. Для этого в системе предусмотрены специальные средства, включенные в выходной язык и обеспечивающие стандартизацию выходных сообщений. В блоке 4 задействованы функции коррелятора, экстраполятора, а также выходного преобразователя.

Изменение состояния в динамике (блок 5) определяется с помощью качественных оценок типа «улучшение», «без изменений», «ухудшение», а темпы изменения — с помощью понятий «незначительное», «значительное», «резкое». Вывод

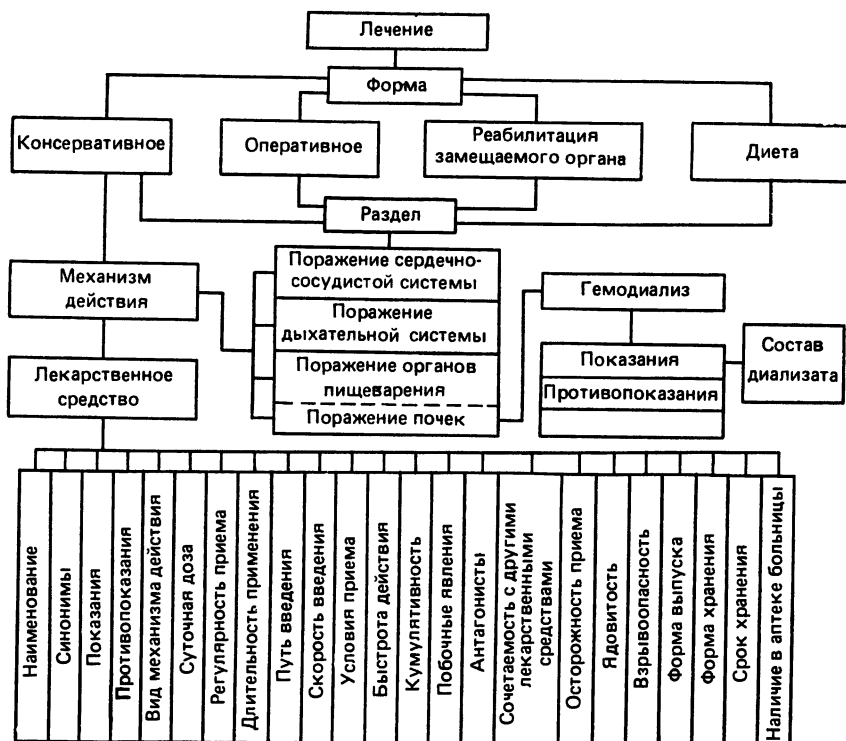


Рис. 18.7. Принципы структурирования лечебных мероприятий

этих оценок производится при наличии в диагнозе сведений о степени тяжести в различные моменты времени, сформированных на основании сочетаний различных градаций большого количества признаков и, таким образом, на высоком с точки зрения абстракции лингвистическом уровне (функции «Коррелятора», «Адаптатора»).

«Лечение» — основная процедура блока 6. Это процедура сложной структуры, манипулирующая данными из МИБ (см. блок 8). В МИБ находятся все сведения о больном, как полученные извне, так и порожденные системой, и из описания лечебных мероприятий, структура которого идентична описанию в виде дерева (рис. 18.7), признаков и патологических состояний. Лечебные мероприятия имеют такие характеристики: раздел, форма, показания механизма действия, синоним, суточная доза, регулярность приема, длительность применения, путь введения, скорость введения, условия приема, антагонисты, побочные явления, сочетаемость с другими лекарственными средствами, кумулятивность и т. д. Благодаря тому, что мы используем многоуровневый язык представления знаний, процедура рекомендации лечения учитывает различные вариации значений данных на разных уровнях и, таким образом, способна к неоднозначному принятию решений. При определенных показаниях больным с острой почечной недостаточностью назначается гемодиализ (очищение крови) с той прописью состава диализата, которая учитывает индивидуальные особенности больного (функции «Коррелятора»).

18.4.3. База знаний. Как отмечалось выше, при реанимации необходимы знания о наличии или отсутствии наиболее важных патологических процессов, про-

текающих в основных медико-биологических системах и органах. При 30—40 основных патологических состояниях, наблюдаемых при почечных нарушениях, а тем более при 200—300 нарушениях в различных биосистемах список всевозможных сочетаний сведений об их наличии или отсутствии должен состоять из огромного количества элементов. Это ведет к необходимости использовать при диагностике состояний организма модели порождающего типа — модели, позволяющие строить различные патологии на основании исходных данных (а не выбирать их из заданного списка). Для реализации этого подхода структура БЗ должна состоять из знаний по анатомии, нормальной и патологической физиологии и описания клинических форм течения патологии. При этом следует отличать модель диагностическую от лечебной. При наличии исходных данных обе модели могут работать отдельно. Совместная работа возможна при условии, когда выход первой модели полностью согласован со входом второй.

18.4.4. Реализация. Реализация системы выполнялась в три этапа.

1. Формирование ЭС. Этот этап связан с описанием задач, структур знаний и детальной выборкой знаний. Для формирования БЗ необходимо привлечь большое число высококвалифицированных медиков. Как показала практика, этого пока достичь не удается, так как такие специалисты с большой неохотой отдают свое время на формализацию знаний, используемых ими уже на уровне интуиции, или вовсе на это не идут. Поэтому разработчикам ЛЕДИ-Z — инженерам, математикам — пришлось (постепенно превращаясь в инженеров по знаниям) самостоятельно обработать большое количество медицинской литературы и истории болезней, после чего только на формальном уровне выборка и структурирование знаний (как декларативных, так и процедурных) обсуждались и согласовывались со специалистами-медиками. В основном для выявления знания врачей использовался метод прямого опроса уникального эксперта.

Помимо заполнения БЗ трудности связаны с вопросом, как представлять эти знания. Способ структурирования зависит от типа задачи, а также от поиска информации в БЗ и организации диалога. Наиболее приемлемой со всех точек зрения оказалась иерархическая организация на семантических сетях с продукционной системой вывода.

2. Переориентирование прикладной модели на язык математических моделей и задач. Единственным методом, подходящим для составления модели решения взаимосвязанных лечебно-диагностических и прогностических задач, является метод ситуационного управления большими системами. В этом случае при описании объектов и ситуаций используются языки RX-кодов и синтагматических цепей. Модель представляет собой совокупность четырех множеств: x^0 — множество базовых понятий; R — множество базовых отношений; $x^1, x^2, \dots, x^n$ — множество производимых понятий; P — множество правил образования производных понятий. Различные варианты формирования производных понятий реализуются процедурой обобщения, в результате выполнения которой производится классификация понятий и структур.

В концептуальной модели около 370 медицинских признаков (включая анамнестические данные, жалобы и данные осмотра, лабораторных исследований и функциональные данные), а со свойствами — около 1000, патологических состояний различной степени детализации — около 200. Программная реализация на примере диагностики острой почечной недостаточности с сопутствующими нарушениями гомеостаза включает примерно пятую часть объема концептуальной модели. Система открытая, т. е. позволяет совершенствовать ее работу, увеличивать полноту и точность знаний, декларативных и процедурных. База знаний ЛЕДИ-Z может послужить основой для создания системы диагностики широкого спектра терапевтических заболеваний, выработки советов врачу в подборе лечения с учетом большого количества факторов. Ее можно использовать как справочник для обучения студентов, как систему для повышения квалификации врачей, как архив историй болезней для выбора статистических данных, их обработки, для учета лекарственных средств и других целей.

3. Программная реализация. Программная реализация осуществлена на языках ФРЛ (ЛИСП) ЕС МЭИ. Основой языка ФРЛ, представляющего собой

средство для работы с процедурно-декларативными сетями, является фрейм, который может использоваться как для описания элементов знания, так и для их представления. ЛИСП позволяет работать с символьной, сложно-структурированной информацией большого объема.

В разработанной системе три типа фреймов, представляющих знания: заносимые сразу в долговременную память (фрейм нормативно-справочной информации ИСТ-БОЛ, фреймы патологических состояний, медицинских признаков, лечебных мероприятий), в кратковременную память (фреймы входных историй болезни) и заносимые из кратковременной памяти в долговременную после их дополнения и обработки (фреймы, формирующие машинный архив ИБ).

Интерфейс с пользователем осуществляется на языке запросов, основанном на сравнении образцов.

В общей сложности для лечебно-диагностической системы создано порядка 100 системных фреймов, один глобальный ИСТ-БОЛ, содержащий сведения о 57 медицинских признаках, 30 процедур обработки фреймов. Требуемая оперативная память для ЛИСП и ФРЛ 125 Кбайт. Объем БЗ на нынешнем этапе разработки 140 Кбайт внешней памяти, которая будет постепенно наращиваться до 15 Мбайт. На одну историю болезни требуется около 3 Кбайт внешней памяти.

В настоящее время завершается реализация ЛЕДИ-Z на ПЭВМ IBM PC/XT, ЕС-1840, ЕС-1841.

СПИСОК ЛИТЕРАТУРЫ

- Автоматизация, 1986 Автоматизация обработки сложной графической информации. - Горький: Горьковский ун-т, 1986. - 198 с.
- Автоматизированные, 1986 Автоматизированные системы обработки изображений: Тезисы докл. 2-я Всесоюз. конф.-М.:Наука, 1986.
- Агаев и др., 1984 Агаев А.А., Джафаров Э.М., Долгополов Б.Я. Банк данных автоматизированной системы обработки изображений // Обработка изображений и дистанционные исследования. / ВЦ СО АН СССР. - Новосибирск, 1984.- С. 55-57.
- Александров и др., 1984а Александров В.В., Арсентьева А.В. Информация и развивающиеся структуры. - Л.: ЛНИВЦ АН СССР, 1984, 185 с.
- Александров и др., 1984б Александров В.В., Фазылов Ш.Х. Интегрированный подход к анализу данных//Автоматизация исследований на основе информационно-вычислительной сети.-Л.: ЛНИВЦ АН СССР,1984.- С. 3-10.
- Александров и др., 1985 Александров В.В., Горский Н.Д. Представление и обработка изображений. Рекурсивный подход. - Л.: Наука, 1985.- 192 с.
- Александров и др., 1986 Александров В.В., Горский Н.Д., Мысько С.Н. Экспертные системы анализа изображений. - Л.: ЛИИАН, 1986.- 37 с.
- Александров В.В. и др. 1987 Александров В.В., Горский Н.Д. Базы видеоданных: состояние и тенденции развития // Прикладная информатика.- 1987.-Вып. 1(12).- С. 24-52.
- Алексеев и др., 1983 Алексеев А.С. и др. Функциональное математическое обеспечение обработки изображений.-Препринт. - Новосибирск, 1983. - 24 с. - (ВЦСО АН СССР, № 410).
- Аллен и др., 1986 Аллен Дж.Ф., Парро Р. Выявление коммуникативного намерения, содержащегося в высказывании //Новое в зарубежной лингвистике. - М.: Прогресс, 1986. - Вып. 17. - С. 322-362.
- Ангелова и др., Ангелова Г., Кох Д., Нариньяни А. Системы общения с ЭВМ на естественном языке //Прикладные человеко-машинные системы, ориентированные на знания. Представление знаний в человеко-машинных и робототехнических системах. - М.:ВИНИТИ, 1984.-Т. С. - С. 36-69.
- Андреев, 1979 Андреев В.П. О возможности построения технической зрительной системы на основе вычислительной среды параллельного действия // Методы и модели для управления роботами и манипуляторами в производстве и научных исследованиях / МНДП; - М., 1979.- С. 76-82.

- Андреев, 1981 Андреев В.П. Ввод телевизионных изображений в ЭВМ и их сегментация // Проблемы машинного видения в робототехнике / Под ред. Д.Е. Охоцинского. - М.: Ин-т прикл. математ. АН СССР, 1981. - С. 152-164.
- Андреев и др., 1982 Андреев В.П., Соколов С.М. Устройство сопряжения промышленной телевизионной установки с ЭВМ и его программное обеспечение. - М. 1982. - 30 с. - Препринт. - (Ин-т прикл. матем. АН СССР, No 181).
- Андреев и др., 1987. Андреев В.П., Белов Д.А., Вайнштейн Г.Г. Эксперименты с машинным зрением. - М.: Наука, 1987. - 128 с.
- Апресян, 1975 Апресян Ю.Д. Лексическая семантика. - М.: Наука, 1975. - 367 с.
- Апресян и др., 1978 Апресян Ю.Д., Богуславский И.М., Иондин Л.Л. Лингвистическое обеспечение в системе автоматического перевода третьего поколения. - М.: Наука, 1978. - 172 с.
- Апресян и др., 1983 Апресян Ю.Д. и др. Лингвистическое обеспечение системы французо-русского автоматического перевода ЭТАП-1. - Препринт. - М., 1983-1985. - (ИРЯ АН СССР, No 154, 155, 159, 160, 167).
- Апресян и др., 1988 Апресян Ю.Д. и др. Лингвистическое обеспечение системы ЭТАП-2. - М.: Наука, 1988
- Арнолд, 1986 Арнолд Д. Eurotra : Будущее машинного перевода в Европе // ТИИЭР. - 1986. - Т. 74, No 7. - С. 98-112.
- Афанасьев и др., 1984 Афанасьев В.П. и др. Архитектура видеоречевого терминала "МАРС-1" // Автоматическое распознавание слуховых образов: Тезисы докл. и сообщ. 13-й Всесоюз. школы-семинара (АРСО-13). Ч.1. - Новосибирск, 1984. - С. 31-32.
- Афанасьев и др., 1986 Афанасьев А.Н. и др. Автоматизированный экспресс-анализ изображений при геологической разведке дна океана // Автоматизированные системы обработки изображений. - М.: Наука, 1986. - С. 181-182.
- Байков и др., 1986 Байков А.М., Власенко А.В., Леонова А.Е., Шанис А.Л. Распознавание рукописной символьной и графической информации в ЭВМ // Вопросы радиоэлектроники. Сер. ЗВТ. - 1986. - No 3. - С. 47-58.
- Баллард и др., 1987 Баллард Д.Х., Браун К.М. Зрение: биология бросает вызов технике // Реальность и прогнозы искусственного интеллекта / Под ред. В.Л. Стефанюка. - М.: Мир, 1987. - С. 103-123.
- Баскин и др., 1987 Баскин И.А., Бебчук С.А., Винц В.М. Инструментальный комплекс для разработки документно-ориентированных информационных систем на персональных ЭВМ // Вопросы синтеза программ (разработка информационных систем / Под ред. Э.З. Любимского. - М.: МГУ, 1987. - С. 81-93.

- Белоногов и др.,
1984 Белоногов Г.Г., Дуганова И.С., Кузнецов А.Б. Экспериментальная система автоматизированного обнаружения и исправления ошибок в тексте//НТИ. Сер 2. - 1984.- N 3. - С. 20-22.
- Бельская, 1969 Бельская И.К. Язык человека и машина.- М.: МГУ. - 1969.
- Богоино и др.,
1986 Богоино В.И., Коряков И.В., Шаледко В.В. Синтез речи в персональных ЭВМ // APCO-14. - Каунас, 1986. - С. 61-63.
- Брябрин,
1981 Брябрин В.М. Адаптивный диалог - основа персональной вычислительной системы // Лингвистические процессоры и представление знаний/ ВЦ СО АН СССР. - Новосибирск, 1981.- С. 22-40.
- Брябрин и др.,
1983 Брябрин В.М., Любарский А.Я., Микулич Л.И. и др. Диалоговые системы в АСУ.//Под ред. Д.А.Попелова. - М.: Энергоатомиздат, 1983.- С. 85-120.
- Бухтилов и др.,
1970 Бухтилов Л.Д., Дегтярев Н.П., Лобанов В.М. Об автоматическом выделении из речевого сигнала акустического признака "тон - не тон" //Тр. Акуст. ин-та. -1970.-Вып. 12. - С. 12-20.
- Бухтилов и др.,
1986 Бухтилов Л.Д., Лобанов В.М. Алгоритм оценки формантных частот // APCO-14. Ч.1.- Каунас, 1986. - С. 10-11.
- Бюллетень, 1988 Бюллетень 1988. Бюллетень иностранной коммерческой информации. Об использовании экспертных систем в ФРГ, №60, 1988, 26 мая, с.4.
- Ван Дан, 1986 Ван Дан А. Машинная графика// Современный компьютер.-М.: Мир, 1986.- С. 108-127.
- Варченко и др.,
1986 Варченко Л.В., Раков М.А., Тонин Ю.А. Архитектура современных систем интерактивной обработки изображений//Автоматизированные системы обработки изображений.- М.: Наука, 1986.- С. 250-252.
- Васильев и др.,
1983 Васильев М.В., Грицаненко А.В., Заслонко А.Н. Адаптивный лингвистический транслятор. // Разработка и применение лингвистических процессоров.- Новосибирск: ВЦ СО АН СССР, 1983.- С. 3-17.
- Вильштер и др.,
1986 Вильштер В., Кобза А. Модели пользователя в диалоговых системах // ТИИЭР.- 1986.-Т. 74, No 7.- С. 62-76.
- Виноград,
1976 Виноград Т. Программа, понимающая естественный язык.- М: Мир, 1976. -294 с.
- Виноград,
1983 Виноград Т. К процессуальному пониманию семантики // Новое в зарубежной лингвистике.- М.: Радуга, 1983.- С. 123-170.
- Винцок,
1987 Винцок Т.К. Анализ, распознавание и интерпретация речевых сигналов. - Киев: Наукова думка, 1987. - 262с.
- Винцок и др., Винцок Т.К., Лобанов В.М., Шинкаж А.Г. Система

- 1982 распознавания речи и система речевого диалога
СРД "Речь-1" на основе микро-ЭВМ // Тезисы докл.
АРС0-12.- Киев, 1982. -С. 516-521.
- Войнов и др.,
1987 Войнов В.К., Влехан М.С. Состав, методики разра-
ботки и эффективность действующих систем машин-
ного перевода: Учебное пособие. -Харьков:
Харьковский ун-т, 1987. -60 с.
- Вудс В.А.
1978 Вудс В.А. Сетевые грамматик для анализа
естественных языков // Кибернетический сборник.
Новая серия.- М.:Мир,1978.-Вып. 13.- С. 120-158.
- Вулф, 1987 Вулф А. Универсальный комплекс для разработки ЭС
реального времени//Электроника,1987,№6.-С.43-46.
- Гардан и др.,
1987 Гардан И., Люка М. Машинная графика и автомати-
зация конструирования.-М.: Мир, 1987.- 270 с.
- Гвида и др.,
1981 Гвида Д.Я., Сокольско М. Общение с системами
искусственного интеллекта на естественном языке
(проект DONAU) // Лингвистические процессоры и
представление знаний.-ВЦ СО АН СССР, Новоси-
бирск, 1981.- С. 41-65.
- Гилей,1981 Гилей В. Интерактивная машинная графика.- М.:
Мир, 1981.- 380 с.
- Гинмельфарб,1985 Гинмельфарб Г.Л. Аппаратные средства и особенно-
сти программного обеспечения диалоговой цифро-
вой обработки изображений //Зарубежная радиоэ-
лектроника.- 1985.-N 10. -С. 87-128.
- Гинмельфарб и др.,
1985 Гинмельфарб Г.Л. и др. Программное обеспечение
цифровой обработки изображений на малых ЭВМ се-
мейства СМ-4// Интеллектуальные роботы и рас-
познавание образов./ИК АН УССР. - Киев, 1985.-
С. 39-49.
- Голиков и др.,
1985 Голиков К.П., Педанов И.Е. РЕДГРАФ: интерактивная
система подготовки чертежно-конструкторской до-
кументации// Тезисы докл. 2-й Всесоюз. конф.-
Горький* Горьковский ун-т, 1985.- С. 207-208.
- Голубев и др.,
1987 Голубев С.А., Гукова С.М., Курашев Е.П., Осипов
Г.С. Пустая система для диагностики технических
объектов // Технология разработки экспертных
систем. - Кишинев, 1987. -С. 52-54.
- Горин,1984 Горин С.В. Графическая система ГРАН-средство
построения программного обеспечения, настраи-
ваемого на правила оформления результатов// Про-
граммные средства машинной графики./ ВЦ СО АН
СССР, Новосибирск, 1984.- С. 17-21.
- Городецкий
и др., 1984 Городецкий В.Ю., Кобозева И.М., Сабурова И.Г.
К типологии коммуникативных неудач. // Диалого-
вое взаимодействие и представление знаний.- ВЦ
СО АН СССР. - Новосибирск, 1985. -С. 64-78.
- Горский, 1986 Горский Н.Д. Системы обработки видеoinформации:
тенденции развития //Методы и системы автомати-

- защи в задачах науки и производства.-М.:Наука, 1986.-С. 151-166.
- Горский, 1989 Горский Н.Д. Восприятие двумерных изображений.- М.: Наука, 1989.
- Гулько, 1987 Гулько Д.Е. Экспертная система для планирования нефтеперерабатывающего производства // Технология разработки экспертных систем. - Кишинев, 1987. -С. 62-64.
- Дегтярев и др., 1987 Дегтярев Н.П., Лесков Е.Я. Повышение надежности распознавания слов слитной речи // Proc. 11th ICPhS. -Tallin, 1987. -Р. 7.
- Делоне, 1959 Делоне Б.Н. Комментарии к книге Гаусса К.Ф. "Труды по теории чисел".- М.: Изд-во АН СССР, 1959.
- Деннинг и др., 1984 Деннинг В.,Эосинг Г., Маас С. Диалоговые системы "человек - ЭВМ". Адаптация к требованиям пользователя.- М.: Мир, 1984.- 112 с.
- Диалоговая, 1987 Диалоговая консультирующая система КОНС-ПРОЛОГ/ Межотраслевая выставка ПРОГРЕСС. - М.: ЦНИИАтоминформ, 1987. - 2 с.
- Дородницын,1985 Дородницын А.А. Информатика: предмет и задачи // Кибернетика. Становление информатики.- М.: Наука, 1985.- С. 22-28.
- Дракин и др., 1988 Дракин В.И., Попов Э.В., Преображенский А.В. Общение конечных пользователей с системами обработки данных.- М.: Радио и связь, 1988. -288 с.
- Дьюдни, 1987 Дьюдни А.К. О фрактальных горах,графтальных растениях и др. графических чудесах фирмы PIXAR// В мире науки.- 1987.- No 2, С. 104-109.
- Закревский, 1982 Закревский А.Д. Выявление имплицативных закономерностей в булевом пространстве признаков и распознавание образов // Кибернетика.- 1982.- No 1.-С. 1-6.
- За рубежом, 1988 За рубежом.- 1988.-N 33.- С.20-21.
- Звягинцев, 1976 Звягинцев В.А. Предложение и его отношение к языку и речи.-М.: МГУ,1976.
- Зенкин, 1983 Зенкин А.А. Обобщение теоремы Гильберта-Варинга // Вестник МГУ. Сер. 1. - 1983.- No 2.- С.11-19.
- Зенкин, 1987 Зенкин А.А. Интерактивная компьютерная графика в теоретических исследованиях: обобщение классической теоремы Гильберта-Варинга // Изв. АН СССР. Сер. Техн. кибернетика.-1987.- No 5.
- Зиновьева, 1984 Зиновьева Н.В. Особенности фонетической интерпретации "слепых" сонোগрам при работе с временным окном // APCO-13. - Новосибирск, 1984. -С. 30-32.
- Зиновьева и др., 1986 Зиновьева Н.В., Лобанов Б.М., Рыжиков В.В. Выбор оптимального параметрического представления сиг-

- налов для построения экспертных систем распознавания речи // АРСО-14. - Ч.1. Каунас, 1986. - С. 67-68.
- Златоустова Л.В., Потапова Р.К., Трунин-Донской В.Н. Общая и прикладная фонетика. -М.:МГУ, 1986. - 304 с.
- Златоустова Л.В., Потапова Р.К., Трунин-Донской В.Н. Общая и прикладная фонетика. -М.:МГУ, 1986. - 304 с.
- Иващенко и др., 1987 Иващенко К.И., Клыкков А.Ю., Маслов С.В. Принятие решений в диалоговой экспертной системе. // Сб. Техника, экономика, информация. Сер. АСУ,-М., ВИМИ, 1987. - С. 75-85.
- Изделие ЭКСПЕРТ, 1988 Изделие ЭКСПЕРТ, АТЛ.00071-01 31 01. Описание применения., 1988.
- Изделие ЭКСПЕРТИЗА, 1988 Изделие ЭКСПЕРТИЗА АТЛЗ.060.6088.Т0 .Техническое описание. 1988.
- Кадзи и др., 1985 Кадзи Х., Окадзина А. Хитати ни окэру кикай хонъяку сисутэму.- Дзехо сери, 1985.- Т. 26, No 10.- С. 1214-1216.
- Катков и др., 1984 Катков В.Л., Хлебцевич Г.Е., Влизнец А.И. РИС-язык наборной графики// Программные средства машинной графики / ВЦ СО АН СССР, - Новосибирск, 1984.- С. 41-51.
- Кейтер, 1985 Кейтер Дж. Компьютеры - синтезаторы речи: Пер. с англ. - М.: Мир, 1985. -237 с.
- Кирсанов и др. Кирсанов В.С., Киселев А.Ф., Попов Э.В. Создание гибридных интеллектуальных систем на основе средств ДИС и ДИЗС // Технология разработки интеллектуальных систем /- М.: МДНТП, 1989.
- Козьмина Е.Л.и др. 1972 Козьмина Е.Л.,Коровина Т.И. Морфологический синтез в системе французско-русского машинного перевода.- Препринт.- М., 1972.- (Ин-т прикладн. АН СССР, No 7).
- Кузин и др.,1982 Кузин М.И., Харнас С.М. Местное обезболивание.- М.: Медицина, 1982.
- Коов, 1987 Коов М.И. Инструментальное средство для построения гибридных экспертных систем// Технология разработки экспертных систем. - Кишинев, 1987. -С. 85-87.
- Кулагина, 1973 Кулагина О.С. О машинном переводе текстов на естественных языках //Проблемы кибернетики.- 1973.- No 27.
- Кулагина, 1979 Кулагина О.С. Исследования по машинному переводу. - М: Наука, 1979. -320 с.
- Кучеров и др., 1983 Кучеров В.Я., Лобанов В.М. Синтезированная речь в системах массового обслуживания. -М.: Радио и связь, 1983. - 132 с.
- Лейкина и др., 1966 Лейкина В.М. и др. Система автоматического перевода, разрабатываемая в группе математической лингвистики ВЦ ЛГУ //НТИ.-1966.- No 1.-С.

- 40-50.
- Леонтьева и др., 1980
Леонтьева Н.Н., Никогосов С.Л. Система ФРАП и проблема оценки качества автоматического перевода // Машинный перевод и прикладная лингвистика. - М., 1980. - Вып. 20. - С. 57-78.
- Леонтьева 1985
Леонтьева Н.Н. Информационная модель системы автоматического перевода // НТИ. Сер. 2. - 1985. - No 10.
- Лобанов, 1982
Лобанов Б.М. Устройство речевого ввода информации "Фонемофон-3" для ЭВМ // Управляющие системы и машины. - Киев: Наукова Думка, 1982. - No 2. - С. 33-37.
- Лобанов, 1982
Лобанов Б.М. Оптимальный модуль синтеза речи по тексту // APCO-12. - Киев, 1982. - С. 516-521.
- Лобанов и др., 1986
Лобанов Б.М. и др. Одноплатный модуль синтеза речи по тексту. // APCO-14. - Каунас, 1986. - С. 54-55.
- Лобанов, 1988
Лобанов Б.М. Состояние и перспективы разработки речевых устройств для интеллектуальных роботов связи // Электросвязь, 1988. - No 8. - С. 42-46.
- Мадо, 1985
Мадо В.Л. Оптические читающие терминалы: Концепции построения // Интеллектуальные читающие терминалы. - М.: ИНЗУМ, 1985. - С. 5-21.
- Махьюин, 1986
Махьюин К.Р. Порождение текста. Области применения, проблемы, методы // ТИИЭР. - 1986. - Т. 74, No 7. - С. 77-85.
- Мальковский, 1985
Мальковский М.Г. Диалог с системой искусственного интеллекта. - М.: МГУ, 1985. - 214 с.
- Мануэль, 1985а
Мануэль Т. Попытки внедрения экспертных систем и проблема интеграции // Электроника. - 1985. - No 14. - С. 31.
- Мануэль, 1985б
Мануэль Т. Что тормозит внедрение экспертных систем? // Электроника, - 1985, - No 16. - С. 18-23.
- Маркин и др., 1987
Маркин В.И., Цыцаркин Ю.М. Графическая диалоговая подсистема в САПР вычислительных устройств, работающих в режиме распознавания образов // Информатика-87 / АН Армянской ССР. - Ереван, 1987. - С. 243-244.
- Марчук, 1983
Марчук Ю.Н. Проблемы машинного перевода. - М.: Наука, 1983. - 233 с.
- Мацокин и др., 1984
Мацокин А.М., Упольников С.А. ППП SPACE-средство формирования и обработки трехмерных объектов // Программные средства машинной графики / ВЦ СО АН СССР. - Новосибирск, 1984. - С. 77-90.
- Машинный, 1987
Машинный перевод и прикладная лингвистика. Проблемы создания системы автоматического перевода. - М., 1987. - 152 с. - (Научные труды МПНИИ. Вып. 271).
- Международный,
Международный семинар по машинному переводу: Те-

- 1983
Международная,
1985
Мельчук, 1974
Методы, 1985
Миран и др., 1982
Мицубиси, 1987
Мураки, 1985
Нагао и др.,
1986
Нечаев, 1953
Новое в лингвистике, 1983
Обработка, 1984
Овсеевич и др.,
1985
Окаси и др.,
1986
Ояхман, 1986
Павлидис, 1986
Падучева,
- зины докл. М.: ВЦП, 1983.- 268 с.
Международная конференция "Теория и практика научно-технического перевода" : Тезисы докл. - М.: ВЦП, 1985.
Мельчук И.А. Опыт теории лингвистических моделей. "Смысл <-> Текст". - М.: Наука, 1974.
Методы и средства обработки сложной графической информации: Тезисы докл. 2-я Всесоюз. конф.- Горький: Горьковский ун-т, 1985.-346 с.
Миран Г.З., Гальченко О.Н. Синтаксический и семантический анализ-синтез в системе машинного перевода СИМПАР //Всесоюз. конф. "Совершенство перевода научно-технической литературы и документов".- М.: ВЦП, 1982.-С. 190-191.
AI ваку сутзоен MELCOM PSI // Мицубиси данки гихо.- 1987. -Т.61, No 1.-С. 47-48.
Мураки К. Нития соохо хоньяку оиоту Venus HO озихэй оисо // Дзехо сери.- 1985.-Т. 26, No 10.-С. 1220-1222.
Нагао М., Цудзими Дз., Наканура Дз. Машинный перевод с японского языка на английский //ТИ-ИЗР.-1986.-Т. 74, No 7.-С. 112- 133.
Нечаев В.И. О представлении натуральных чисел оуинной слагаемых вида $x(x+1)...(x+k-1)/k!$ // Изв. АН СССР. Матем.- 1953.-Т. 17, No 6.-С. 485-498.
Новое в зарубежной лингвистике.- Прикладная лингвистика/Соот. В.А.Звягинцев. -М.: Радуга, 1983.- Вып. 12.-214 с.
Обработка изображений и дистанционные исследования.Т.2.Технические средства обработки изображений и автоматизации проектирования: Тезисы докл. 2-я Всесоюз. конф. - Новоосибирск:ВЦ СО АН СССР, 1984.- С. 84.
Овсеевич И.А., Ярославский Л.П. Автоматизированные системы обработки изображений. Состояние и перспективы//Материалы 1-й Международной школы по автоматизации научных исследований.- Пушкино: Научный центр биологических исследований АН СССР, 1985.- С. 278-286.
Окаси М., Судзуки К., Оно Х. Нития хикай хоньяку оиотуэн Thalia-3 //Мицубиси данки гихо.-1986.-Т. 60, No 3.-С. 37-40.
Ояхман Е.Г. Графические системы для СМ ЭВМ.- М: Наука, 1986.- 192 с.
Павлидис Т. Алгоритмы машинной графики и обработки изображений.- М.: Финансы и статистика, 1986.-110 с.
Падучева Е.В. Актуализация предложения в составе

- 1982а речевого акта // Формальное представление лингвистической информации.- Новосибирск: ВЦ СО АН СССР, 1982.-С. 38-63.
- Падучева, 1982б Падучева Е.В. О семантике синтаксиса. -М.: Наука, 1982.-360 с.
- Певзнер 1981 Певзнер Б.Р. Система машинного перевода с элементами обучения //Вопросы информационной теории и практики /ВИНИТИ.-М.,1981.-№ 46. -С. 104-110.
- Петров и др., 1980 Петров и др. Система ввода речевых сигналов для ЭВМ // АРСО-11.-Ереван, 1980. -С. 280-282.
- Пизони и др., 1985 Пизони Д.В., Мусбаум Г.С., Грин В.Г. Восприятие синтезированной речи, генерируемой по правилу // ТИИЭР.- 1985.- N 11. -С.146-160.
- Попов и др., 1976 Попов Э.В., Фирдиан Г.Р. Алгоритмические основы интеллектуальных роботов и искусственного интеллекта. -М.: Наука, 1976.-324 с.
- Попов и др., 1978 Попов Э.В., Родионов Е.В. Алгоритм синтеза предложений ограниченного русского языка из семантической сети //Изв. АН СССР.Техн. кибернетика. - 1978.-№ 6.-С. 44-51.
- Попов, 1982 Попов Э.В. Общение с ЭВМ на естественном языке. -М.: Наука, 1982. -360 с.
- Попов, 1987 Попов Э.В. Экспертные системы. Решение неформализованных задач в диалоге с ЭВМ.-М.: Наука, 1987. -288 с.
- Попов и др., 1988 Попов Э.В., Киселев А.Ф., Орлов А.А. Инструментальные средства построения ИПС для персональных ЭВМ // Материалы семинара "Развитие интеллектуальных возможностей современных и перспективных ЭВМ". - М.:МДНТП, 1988,-С. 57-66.
- Поспелов и др., 1972 Поспелов Д.А., Пушкин В.Н. Мышление и автоматы. - М.:Сов. радио, 1972. - 224с.
- Поспелов, 1988 Поспелов Г.С. Искусственный интеллект - основа новой информационной технологии // Коммунист. 1988.-№ 1.-С. 88-96.
- Пранц 1983 Пранц М.Г. ТАСС - диалоговая система обработки текста // Разработка и применение лингвистических процессоров/ ВЦ СО АН СССР.- Новосибирск, 1983.-С. 37-43.
- Представление, 1984 Представление знаний в человеко-машинных и робототехнических системах/ВИНИТИ. -М.; 1984.
- Преображенский, и др., 1979 Преображенский А.В., Рыбина Г.В., Хорошевский В.Ф. Генерация многоцелевых интеллектуальных вопросов-ответных систем //Изв. АН СССР. Техн.кибернетика.-1979.-№ 6.-С. 142-151.
- Преображенский, 1982 Преображенский А.В. Лингвистический процессор для реализации общения с базами данных //Изв. АН СССР Техн. кибернетика.-1982.- № 5.-С.121-129.
- Рыбина, Рыбина Г.В. Естественно-языковая система для ав-

- 1982 томатизации расчетных работ в современных САПР и АСУ ТП // Прикладные и экспериментальные лингвистические процессоры / ВЦ СО АН СССР. - Новосибирск, - 1982.-С. 60-75.
- Сазонов, 1985 Сазонов К.А. Графический интерактивный синтез трехмерных моделей в системе ИНТЭАР// Тезисы докл. 2-я Всесоюз. конф.- Горький: Горьковский ун-т.-1985.-С. 272-273.
- Сансонов, 1987 Сансонов В.В., Эксперимент по реализации ЭС Консультант-2 методом трансляции базы знания из глубинного представления в поверхностное// Технологическая разработка экспертных систем. - Кишинев, 1987. -С. 116-120.
- Сапожков, 1963 Сапожков М.А. Речевой сигнал в кибернетике и связи. - М.: Связьиздат, 1963. -472 с.
- Сапожков и др., 1983 Сапожков М.А., Михайлов В.Г. Вокодерная связь.- М.: Радио и связь, 1983. -247 с.
- Синтаксический, 1981 Синтаксический компонент в системах машинного перевода. - М., 1981. - 132 с.
- Система ДИСОД, 1987 Система управления базой данных ДИСОД /Е.С.Броневских, В.И.Бурдаков, Л.И.Гуков и др. /Под общ. рук. В.И.Дракина. -М.: Финансы и статистика, 1987.-263 с.
- Слуцкер, 1968 Слуцкер Г.С. Нелинейный метод анализа речевых сигналов //Труды НИИР. -1968.-Вып. 2.-С. 32-41.
- Смирнов и др., 1984 Смирнов Г.И., Зиницкий Ю.А., Конушанов А.А., Применение систем распознавания и синтеза речи// АРСО-13. - Новосибирск, 1984. - С. 168-169.
- Соловьев и др., 1985 Соловьев С.Ю., Соловьева Г.М. Предоставление знаний в системах альтернатив // Логико-комбинаторные методы в искусственном интеллекте и распознавании образов.- Кишинев, 1985.-С. 16-20.
- Соловьев и др., 1986 Соловьев С.Ю., Соловьева Г.М. Инструментальная экспертная система ФИАКР. - ГосФАП, Ном.гос.рег. 50860001141 от 26.11.86, 300 с.
- Соловьева, 1985 Соловьева Г.М. Предоставление знаний в системах альтернатив.-Препринт /ИМ с ВЦ АН МССР, 1985. -24 с.
- Танака и др., 1985 Танака Х., Мидзогути Ф. Ини сери о доню сита кикай хоньяку ни цуэтэ. Юго хосики //Дзехо сери.-1985.-Т. 26, No 10.-С. 1191-1196.
- Тихониров, 1984 Тихониров В.Д. Некоторые особенности программного обеспечения и технологии систем машинного перевода АМПАР и НЕРПА //Международный форум по информации и документации.- 1984.-Т. 9, No 2.-С. 8-10.
- Тихониров, 1987 Тихониров В.Д. Промышленные системы машинного перевода // Научно-технический перевод.- М.: Наука, 1987.- с. 92-105.

- Томпсон и др., 1987
Трунин-Донокой, 1984
Трунин-Донокой, 1987
Убар, 1980
Уилкс, 1983
Ульян, 1983
Утида, 1985
Файн, 1985
Фант, 1964
Хорошевский, 1986
Хуа Ло-ген, 1964
Циннан, 1986
Шаляпина, 1974
Шаляпина, 1977
Шаляпина, 1980
- Томпсон З., Томпсон У. Анатомия экспертной системы // Реальность и прогнозы искусственного интеллекта. - М.: Мир, 1987.
Трунин-Донокой В.Н. Алгоритмы и системы восприятия речи // Предоставление знаний в человеко-машинных и робототехнических системах / ВИНТИ. - М., 1984. - С.70-111.
Трунин-Донокой В.Н. Разработка советских ученых в области автоматического распознавания речи // Актуальные вопросы фонетики в СССР / ВИНТИ. - М., 1987. - С.71-106.
Убар Р.Р. Текстовая диагностика цифровых устройств. - Таллин: Таллин. политехн. ин-т, 1980. - 180 с.
Уилкс Я. Анализ предложений английского языка // Новое в зарубежной лингвистике. - М.: Радуга, 1983. - Вып. 12. - С. 318-372.
Ульян Д. Основы баз данных. - М.: Мир, 1983. - 289 с.
Утида Х. Бун рикай ни мотодзуку кикай хоньяку оисутэцу ATLAS-2 // Дзехо сери. - 1985. - Т. 26, No 10. - С. 1217-1219.
Файн В.С. О машинном понимании естественной речи - М.: ВИНТИ, 1985. - 43 с.
Фант Г. Акустическая теория речеобразования. - М.: Наука, 1964. - 284 с.
Хорошевский В.Ф. Разработка и реализация экспертных систем - инструментальный подход. // Изв. АН СССР. Техн. кибернетика. - 1986. - No 5. - С.105-114.
Хуа Ло-ген. Метод тригонометрических сумм и его применения в теории чисел. - М.: Мир, 1964.
Циннан Л.Л. Язык для записи лингвистической информации в системе автоматического перевода ЭТАП (опыт "практической логики") // Семиотика и информатика / ВИНТИ. - М., 1986. - Вып. 27. - С. 82-120.
Шаляпина З.М. Семантико-синтаксический анализ в системе англо-русского автоматического перевода (АРАП). - М., 1974. - 58 с. - (ИРЯ АН СССР. ПГЭПЛ. Предв. публ. Вып.47).
Шаляпина З.М. Многоаспектный автоматический словарь с грамматическим обеспечением и его использование при автоматическом анализе текстов // Вопросы кибернетики. Формализация языковых данных. 1977. - Вып. 17. - С. 95-115.
Шаляпина З.М. Макет лингвистического обеспечения системы японско-русского автоматического перевода ЯРАП: общая структура и основные компонен-

- ты. -М.: Ин-т востоковедения АН СССР. -1980. -235 с.
- Шенк, 1988 Шенк Р. Обработка концептуальной информации. -М.: Энергия, 1980. -380 с.
- Шенк и др. 1983 Шенк Р., Лебовиц М., Бирнбаум Л. Интегральная познающая система // Новое в зарубежной лингвистике. -М.: Радуга, 1983. -Вып. 12. -С. 401-449.
- AI, 1987 Artificial Intelligence Markets. -1987. - Vol. 3, N 3-5.
- Addis, 1980 Addis T.R. Expert Systems: an Evolution Information Retrieval // Int.Computer Limited (ICL) Technical J. -May 1980.
- Aiello et al., 1981a Aiello N., Nii H.P. AGEPUFF: a Simple Event-Driven Program // Rep. HPP-81-25, Computer Science Dept., Stanford University. -1981.
- Aiello et al., 1981b Aiello N., Bock C., Nii H.P. Joy of AGE-ing: an Introduction to the AGE-1 System // Rep. HPP-81-23, Computer Science Dept., Stanford University. -1981.
- Aikins, 1983 Aikins J.S. Prototypical Knowledge for Expert Systems // AI. -1983. -Vol. 20. -P. 163-210.
- Aikins et al., 1983 Aikins J.S., Kunz J.C., Shortliffe E.H. PUFF: an Expert System for Interpretation of Pulmonary Function Data // Computers and Biomedical Research. -1983. -Vol. 16. -P. 199-208.
- Aldridge, 1984 Aldridge J.P. AIRID-an Application of the KAS/Prospector Expert System Builder to Identification // Proc. SPIE, Int. Society for Optical Engineering, Bellingham, Wash., Applications AI. -1984. -Vol. 485.
- Alexander et al., 1984 Alexander J.H., Freiling M.J. Troubleshooting with of an Expert System // Techn. Rep. CR-85-05, AI Dept., Computer Research Laboratory, Tektronix, Inc. -Aug. 1984.
- Allen et al., 1982 Allen J.F., Frish A.M., Litman D.J. ARGOT: the Rochester Dialogue System // Proc. Nat. CAI (AAAI-82). -Pittsburgh, 1982. -P. 66-70.
- Amano, 1986 Amano S. The Teshiba Machine Translation System // Japan Computer Quarterly. -1986. -N 64. P. 32-35.
- Appelo, 1986 Appelo L. A Compositional Approach to the Translation of Temporal Expressions in the Rosetta System // COLING. -1986. -P. 313-318.
- Appelt, 1982 Appelt D.E. Planning Natural Language Utterances // Proc. Nat. CAI. -Pittsburgh, 1982. -P. 59-62.
- Appelt, 1983 Appelt D.E. TELEGRAM: a Grammar Formalism for Language Planning // Proc. 8th IJCAI. -Karlsruhe, 1983. -P. 595-599.

- Apte et al.,
1984a Apte C.W., Weiss S.M. A Framework for Expert Control Interactive Software Systems // Rep. CBM-TR-144, Laboratory Computer Science Research, Rutgers University.-New Brunswick N.J., Oct. 1984.
- Apte et al.,
1984b Apte C.W., Weiss S.M. A Knowledge Representation Framework Expert Control Interactive Software Systems // Rep. CBM-TR-143, Laboratory Computer Science Research, Rutgers University.-New Brunswick N.J., Oct. 1984.
- Baas et al.,
1984 Baas L., Bourne J.R. A Rule-Based Microcomputer System for Electroencephalogram Evaluation // IEEE Trans. -1984.-Vol. BME-31, N 10.
- Bachant et al.,
1984 Bachant J., McDermott J. P1 Revisited: Four Years in the Trenches // AI Magazine.-1984.-Vol. 5, N 3.
- Bajacsy et al.,
1985 Bajacsy R., Joshi A., Krotkov E. A Natural Language and Computer Vision System for Analysing Aerial Images // Proc. 9th IJCAI.- Los Angeles, 18-23 Aug. 1985.-P. 919-921.
- Banares-Alcantara
et al., 1983 Banares-Alcantara P., Westerberg A.W., Rychner M.D. Development of an Expert System for Physical Property Predictions // Report, Design Research Center Robotics Institute, Carnegie-Mellon University.-Pittsburgh, 1983.
- Barth et al.,
1983 Barth S.W., Coyle H., Sobic D. User Guide for the Duplex Army Radio/Radar Targeting Aid // Report, Par Technology Corporation.-New Hartford N.Y., 1983.
- Batali et al.,
1980 Batali J., Hartheimer A. The Design Procedure Language Manual // Rep. AIM-598, AI Laboratory, MIT.-Cambridge (Mass.), Sept. 1980.
- Bates, 1984 Bates M. Accessing a Database with a Transportable Natural Language Interface // Proc. CAI Applications.-Sheraton: Denver Tech. Center, 1984.-P. 9-12.
- Bates et al.,
1984 Bates M., Bobrow R.J. Natural Language Interfaces: What's Here, What's Coming, and Who Needs it // Artificial Applications for Business.-Norwood, NJ: Ablex Publishing, 1984.
- Bayer et al.,
1985 Bayer S., Joseph L., Kalish C. Grammatical Relations as the Basis for Natural Language Parsing and Text Understanding // Proc. 9th IJCAI.- Los Altos (Calif.), 1985.-P. 788-790.
- Bennett et al.,
1978 Bennett J., Creary, Engelmire P. A Knowledge-Based Consultant for Structural Analysis // Computer Science Dept., Stanford University.-Stanford (Calif.), Sept. 1978.

- Bonnet et al., 1983 Bonnet A., Dahan C. Oil-Well Data Interpretation Using Expert System and Pattern Recognition Technique // Proc. 8th IJCAI.-1983.-P. 185-189.
- Boose, 1984 Boose J.H. Personal Construct Theory and the Transfer of Human Expertise // Proc. AAAI.-1984.-P. 27-33.
- Brachman, 1979 Brachman R. On the Epistemological Status of Semantic Networks // Associative Networks: Representation and Use of Knowledge by Computer.- New-York: Academic Press, 1979.
- Bresnan et. al., 1982 Bresnan J., Kaplan R. Lexical-Functional Grammar: a Formal System for Grammatical Representation // Mental Representation Grammatical Relations.-Cambridge (Mass.): MIT Press, 1982.-P. 173-281.
- Briggs, 1982 Briggs F.A. PUMPS Architecture for Pattern Analysis and Image Database Management // IEEE Trans.-1982.-Vol Com-31, N 10.-P 969-983.
- Brinkmann, 1982 Brinkmann K.D. Integration of a Picture-Oriented Data-Based System in an Image Processing System // Proc. 6th Int. Conf. Pattern Recognition, Munich, IEEE.-1982.-P. 798-801.
- Brooks, 1981 Brooks R. Symbolic Reasoning Among 3-Dimensional Models and 2-Dimensional Images // AI-1981.-Vol. 17.-P. 285-349.
- Brooks, 1983 Brooks R. Model-Based 3-Dimensional Interpretation of 2-Dimensional Images // IEEE Trans.-1983.-Vol. PAMI-15.-P. 140-150.
- Brown et al., 1975a Brown J.S., Burton R.R. Multiple Representations of Knowledge for Tutorial Reasoning // Representation and Understanding: Studies in Cognitive Science.- Academic Press.-1975.-P. 311-349.
- Brown et al., 1975b Brown J.S., Burton R.R., Bell A.G. SOPHIE: a Step Toward Creating a Reactive Learning Environment // Int. J. Man-Machine Studies.-1975.-Vol. 7.-P. 675-696.
- Brown et al., 1979 Brown J.S., Burton R.R. Toward a Natural Language Capability for Computer-Assisted Instruction // Procedures for Instructional Systems Development.- New-York: Academic Press, 1979.-P. 273-313.
- Brown et al., 1982a Brown J.S., Burton R.R., DeKleer J. Pedagogical, Natural Language and Knowledge Engineering Techniques in SOPHIE I, II and III.// Intelligent Tutoring Systems.-New York: Academic Press, 1982.-P. 227-282.

- Benett et al., 1981 Benett J., Hollander C.R. DART: an Expert System for Computer Fault Diagnosis // Proc. IJCAI - 1981.-P. 843-845.
- Benett et al., 1984 Benett J., Englemore R. Experience Using EMYCIN.// Rule-Based Expert Systems, Reading.- Addison-Wesley (Mass.), 1984.
- Binford, 1982 Binford T.O. Survey of Model-Based Image Analysis Systems // Int. Journal Robotics Research.-1982.-Vol. 1, N 1.-P. 18-64.
- Blum, 1982 Blum R.L. Discovery, Confirmation and Incorporation of Causal Relationships from a Large Time-Oriented Clinical Database: the RX Project // Computer and Biomedical Research.- 1982.-Vol. 15.-P. 164-187..
- Bobrow et al., 1977a Bobrow D.G., Kaplan R.M., Kay M. GUS: a Frame Driven Dialogue System // AI.-1977.-Vol. 8.-P. 155-173.
- Bobrow et al., 1977b Bobrow D.G., Winograd T. An Overview of KRL, a Knowledge Representation Language // Cognitive Science.-1977.-Vol. 1, N 1.-P. 3-46.
- Bobrow, 1980 Bobrow D.G. Knowledge Representation for Syntactic/Semantic Processing // Proc. Annual Nat. Conf. AI, Stanford Univ.-1980.-P. 316-323.
- Bokhua et al., 1982 Bokhua N.K., Gelovani V.A., Kovrigin O.V. An Expert for Diagnostics of Various Forms of Arterial Hypertension // Tech. Kibern.- 1982.-Vol. 20, N 6.-P. 106-111.
- Boley, 1983 Boley H. FIT-PROLOG: a Functional/Relational Language Comparison // Rep. SEKI-83-14, Fachbereich Informatik, Universitat Kaiserslautern, Postfach 3049, D-6750 Kaiserslautern 1.-West Germany, Dec. 1983.
- Bonasso, 1983 Bonasso R.P. Expert Systems for Intelligence Fusion of the Army Conference on Applications of AI to Battlefield Information Management // Rep. AD-A139 685, Battelle Columbus Laboratories.-Washington: D.C., Apr. 1983.
- Bonasso, 1984 Bonasso Jr.-R.P. ANALIST, an Expert System for Proceeding Sensor Returns // Rep. MTR-83W00002, MITRE Corporation, 1820 Dolly Madison Blvd., McLean, Va.-Febr. 1984.
- Bonissone et al., 1983 Bonissone P.P., Johnson H.E. Expert System for Diesel Electric Locomotive Repair // Knowledge-Based Systems Report, General Electric Co., Schenectady.-N.Y., 1983.
- Bonnet, 1979 Bonnet A. Understanding Medical Jargon as if it were a Natural Language // Proc. 6th IJCAI.-Tokyo, 1979.-P. 79-81.

- Brown et al., 1982 Brown H., Buckman J., Engelmores R. Communication Intelligence Task - HANNIBAL Demonstration // Report ESL, Inc.-Sunnyvale (Calif.), 1982.
- Bruderer, 1982 Bruderer H.-E., ed. Automatische Sprachuber-
setzung.-Darmstadt, 1982.
- Buchanan et al., 1978 Buchanan B.G., Feigenbaum E.A. DENDRAL and
MetaDENDRAL: their Applications Dimension
// AI.-1978.-Vol. 11.
- Byte, 1987a Byte, Jan. 1987.-P. 341.
- Byte, 1987b Byte, Febr. 1987.-P. 44.
- Byte, 1987c Byte, June 1987.-P. 321, 265-272.
- CAIA, 1984 Conf. Artificial Intelligence Applications
(CAIA) 1984.-P. 344, 350, 460, 483, 496, 570, 439,
606, 610.
- Caelen et. al., 1986 Caelen J., Vigonroux N. Producing and Organ-
ising Phonetic Knowledge from Acoustic Facts
in Multi-Level Data-Information // Proc.
ICASSP.-Tokyo, May 23 1986.-P. 1209-1212.
- Callero et al., 1984 Callero M., Waterman D.A. TATR: a Prototype
Expert System for Tactical Air Targeting //
Rep. R-3096-ARPA, Rand Corporation.-June 1984.
- Cantone et al., 1983 Cantone R.R., Pipitone F.I., Lander, W.B. Model-
Based Probabilistic Reasoning Microelectro-
nics Troubleshooting // Proc. 8th IJCAI.-Karl-
sruhe, 1983.
- Carberry, 1983 Carberry S. Tracking User Goals in an Informa-
tion-Seeking Environment // Proc. Nat. Conf.
on Artificial Intelligence.-Washington, 1983.-
P. 59-63.
- Carbonell et al., 1981 Carbonell J.G., Hayes P.G. Dynamic Strategy
Selection in Flexible Parsing // Proc.
19th Annual Meeting Association
for Computational Linguistics.-Stanford
University, 1981.-P. 143-147.
- Carbonell et al., 1983 Carbonell G., Boggs W.M., Mauldin M.L. The
XALIBUR Project: a Natural Language Interface
to Expert Systems // Proc. 8th IJCAI.-
Karlsruhe, 1983.-P. 653-656.
- Cat, 1988 Computers and Translation/ Ed. S. Nirenburg.
N 1.-Pittsburg, 1988.
- Chester, 1976 Chester D. The Translation of Formal Proofs
into English // AI.-1976.-Vol. 7.-P. 261-275.
- Chester et al., 1984 Chester D., Lamb D. Rule-Based Computer Alarm
Analysis in Chemical Process Plants // Proc.
7th Annual Conf. Computer Technolog.
MICRODELCON 84, IEEE.-1984.-P. 22-29.
- Chung et al., 1986 Chung H.-S., Kunii T.-L. NARA: A Two-Way Simu-
ltaneous Interpretation System between Kore-

- an and Japanese. A Methodological Study // COLING.-1986.-P. 325-328.
- Clancey,1979a Clancey W. Tutoring Rules for Guidinga Case Method Dialogue // Int. J. Man-Machine Studies.-1979.-Vol. 11.-P. 25-49.
- Clancey,1979b Clancey W. Dialogue Management for Rule-Based Tutorials // Proc. IJCAI-1979.-P. 155-161.
- Clancey,1981 Clancey W. Methodology for Building an Intelligent Tutoring System // Rep. STAN-CS-81-894,Computer Science Dept.-Stanford University.-1981.
- Clancey,1983a Clancey W. GUIDON // J. Computer-Based Instruction.-1983.-Vol. 10.-P. 8-15.
- Clancey,1983b Clancey W. The Advantages of Abstract Control Knowledge in Expert System Design // Proc. AAAI.- 1983.-P. 74-78.
- Clancey,1983c Clancey W. MRSNEOMYCIN: Representing Meta-Control Predicate Calculus // Rep. HPP-82-31,Computer Science Dept.-Stanford (Calif.),Jan. 1983.
- Clayton et al., Clayton J.,Friedland P. SEQ-Sequence Analysis System // Rep. HPP-81-3,Computer Science Dept., Stanford University and School Medicine. - Stanford (Calif.),1981.
- 1981
- Cohen et al., Cohen P.,Lieberman M.D. A Report on FOLIO: an Expert Assistant for Portfolio Managers // 1983 Proc. 8th IJCAI.-1983.-P. 212-214.
- COLING,1988 COLING: Internat. Conf. Computational Linguistics. -Budapest,1988.
- Colmerauer,1971 Colmerauer A. Les Systems-Q: an Formalisme Pour Analyser et Synthetiser Des Phrases Sur l'Ordinateur // Etudes en Linguistique Appliquee a la Traduction Automatique: etc 1971.-Montreal,1971.-P.1-45.
- Comino et al., Comino R.,Gemello R.,Guida G. Understanding 1983 Natural Language Through Parallel Processing of Syntactic and Semantic Knowledge: an Application to Data Base Query // Proc. 8th IJCAI.-Karlsruhe,1983.-P. 663-667.
- Context,1985 Context Vision-Image Processing System // Lincoping: Context Vision,1985.
- Corey et al., Corey,E.J.,Wipke W.T. Computer Assisted Design of Complex Organic Synthesis // Science.- 1969 1969.-Vol. SMC-12,N 2.-P. 178-192.
- Cullingford et al., Cullingford R.E.,Krueger M.W. Automated Explanations as a Component of a Computer-Aided Design System // IEEE Trans.-1982.-Vol.SMC-12, 1982 N 2.-P. 168-181.
- Darrow,1984 Darrow B. Research Spurs Development of Talk-

- ing Mashines // Design News.-1984.-Vol. 40.
N 12., P. 109-114.
- Datamation,1986 Datamation,1986.-Vol. 32,N 5.
- Davis et al., Davis R.,Austin H.,Sneiderman R. The Dipmet-
1981 er Advision: Interpretation of Geologic Sig-
 nals // Proc. IJCAI.-1981.-P. 846-849.
- Davis et al., Davis R.,Shrobe H.,Hamscher W. Diagnosis Bas-
1982 ed on Description of Structure and Function
 // Proc. AAAI.-1982.-P. 137-142.
- Davis et al., Davis R.,Shrobe H. Representing Structure and
1983 Behavior of Digital Hardware // Proc. IEEE
 Computer.-1983.
- Davis,1976 Davis R. Applications of Metalevel Knowledge
 to the Construction,Maintenanse and Use of
 Lagre Knowledge Bases // Rep. STAN CS-76-
 552,Stanford Laboratory,Stanford Univer-
 sity.-Stanford (Calif.),1976.
- Davis,1978 Davis R. Representation as a Basis for System
 Construction and Maintenance // Pattern-Di-
 rected Inference System.- Academic Press.- 1978.
- Davis,1979 Davis R. Interactive Transfet of Expertise
 Acquisition of New Inference Rules // Artifi-
 cial Intelligence.-1979.-Vol. 12.-P. 121-157.
- Davis,1982 Davis R. Consultation,Knowledge Acquisi-
 tion,and Instruction: Acase Study // AI in Medi-
 cine,Bolider.-Colo.: Westview Press,1982.
 -P. 57-78.
- Deering et al., Deering M.,Faletti J.,Wilensky R. PEARL: an
1981 Efficient Language for AI Programming // Proc.
 7th IJCAI.-Vancouver,1981.
- De Jong,1979 De Jong J. Prediction and Substantiation:
 Two Processes that Comprise Understanding
 // Proc. 6th IJCAI.-Tokyo,1979.-P. 217-222.
- DeKleer,1986 DeKleer J. An Assumption based TMS // Artifi-
 cial Intelligence.-1986.- V.28. -P.127-162.
- Derr et al., Derr M.A.,McKeown K.R. Using Focus to Gene-
1984 rate Complex and Simple Sentence // Proc.
 10th Int. Conf. Computational Linguistics.
 -Stanford,1984.-P. 319-326.
- Dincbas,1980 Dincbas M. A Knowledge-Based Expert Systems
 for Automatic Analysis and Synthesis in CAD//
 IFIPS Proc.-1980.-P. 705-710.
- Drazovich, Drazovich R-J. Sensor Fusion in Tactical War-
1983 fare // AIAA Computers in Aerospace 4 Conf.-
 1983.
- Duda et al., Duda R.,Reboh R. AI and Decision Making: the
1984 PROSPECTOR Expeience // Artificial Intelligence
 Application for Business, Ablex, 1984.
- Dungan,1983 Dungan C-A Model of an Audit Judgment in the

- Form of an Expert System // PhD Dissertation, University of Illinois, Champaign-Urbana, 1983.
- Eiselt, 1985 Eiselt K.P. A Parallel-Process Model of On-Line Inference Processing // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 863-869.
- Engelman et al., 1979a Engelman C., Berg C.H. KNOBS: an Experimental Knowledge-Based Tactical Air Mission Planning System and a Rule Based Aircraft Identification Simulation Facility // Proc. IJCAI.-1979.-P. 247-249.
- Engelman et al., 1979b Engelman C., Allan T. Structure and Function of the Crysallis System // Proc. IJCAI.-1979.
- Eusenat et al., 1985 Eusenat B., Normier B., Ogonowski A. SAP-HFIR+RESEDA, a New Approach to Intelligent Data Base Access // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 855-875.
- Expert Systems, 1983 Expert Systems, 1983.-Vol. 3, N 1.
- Expert Systems, 1984 Expert Systems, 1984.-Vol. 1, N 2.
- Expert Systems, 1985 Expert Systems, 1985.-Vol. 2, N 1, 2, 4.
- Expert Systems, 1986 Expert Systems, 1986.-Vol. 3, N 1-3.
- Expert Systems User, 1987 Expert Systems User, 1987.-Vol. 2, N 10-12.
- Fagan et al., 1979a Fagan L.M., Kunz J.C. Representation of Dynamic Clinical Knowledge: Measurement Interpretation in the Intensive Care Unit // Proc. IJCAI.-1979.-P. 260-262.
- Fagan et al., 1979b Fagan L.M., Kunz J.C. A Symbolic Processing Approach to Measurement Interpretation in the Intensive Care Unit // Proc. 3rd Annual Symp. Computer Applications in Medical Care.-New York, 1979.-P. 30-33.
- Fagan et al., 1984 Fagan L.M., Kunz J.C. Extension to the Rule-Based Formalism for a Monitoring Task Rule-Based Expert Systems, Reading, Mass.- Addison-Wesley, 1984.-P. 397-423.
- Ferguson, 1983 Ferguson G.R. Aircraft Maintenance Expert Systems // Master's Thesis, Air Force Institute of Technology, Wright-Patterson AFB.-Ohio, 1983.
- Ferrante et al., 1984 Ferrante R.D., Carlotto M.J., Pomerode J.M. Multispectral Image Analysis System // Proc. 1st CAI Applications.-Sheraton, 5-7 Dec. 1984.-P. 357-363.
- Finin et al., 1979 Finin T., Goodman B., Tennat H. JETS: Achieving Completeness through Coverage and Closure

- Forgy,1982 // Proc. 6th IJCAI.-Tokio,1979.-P. 275-281.
 Forgy C. RETE: a Fast Algorithm for the Many
 Pattern. / Many Object Pattern Match Problem
 // AI.-1982.-Vol. 19.-P. 17-38.
- Fountain,1986 Fountain T.J. Array Architectures for Iconic
 and Symbolic Image Processing // Proc. 8th
 Int. Conf. Pattern Recognition, IEEE Compu-
 ter Soc. Press.-1986.- P. 24-33.
- Fox et al., Fox M.S.,Allen B. Job-Shop Scheduling an Inve-
 1982 stigation in Constraint-Directed Reasoning //
 Proc. AAAI.- 1982.-P. 155-158.
- Fox et al., Fox M.S.,Allen B. ISIS: a Constraint-Directed
 1983 Reasoning Appmach to Job Shop Scheduling //
 Proc. IEEE Conf. Trends and Applications.-
 1983.
- Foyster,1984 Foyster G. A Knowledge-Based Approach to Tran-
 sistor Sizing // Rep. HPP-84-3,Computer
 Science Dept.-1984.
- Friedman,1983 Friedman L. Research at Jet Propulsion Labo-
 ratory // AI Magazine.-1983.-P. 58-59.
- Friedman,1984 Friedman L. Controlling Production Firing:the
 FCL Language // Report,Jet Propulsion Labo-
 ratory,California Institute of Texnology,Pasa-
 dena.-Caif.,1984.
- Fum et al., Fum D.,Guida G.,Tasso C. Evaluationsg Imper-
 1985 tance: a Step Towards Text Summarization //
 Proc. 9th IJCAI.- Los Altos (Calif.), 1985.
 -P.840-844.
- Gaschnig et al., Gaschnig,J.,Reboh. Development of a Knowl-
 1981 edge-Based Expert System for Water Resurce
 Problems // Rep.SRI-1619,AI Center,SRI In-
 ternational,Menlo Park.-Calif.,1981.
- Gaschnig,1981 Gaschnig J. PROSPECTOR: an Expert System for
 mineral Exploration // Mashine Intelligence,
 Infotech State of the Art Report 9.-1981.-N 3.
- Gaschnig,1982a Gaschnig J. Application of the PROSPECTOR
 System to Geological Exploration Problems //
 Machine Intelligence 10,Chichester,England.
 -Horwood,1982.
- Gaschnig,1982b Gaschnig J. PROSPECTOR: an Expert System for
 Mineral Exploration. // Introductory Readings
 in Expert System,Gordon and Breach,Science
 Publishers.-1982.
- Gazdar et al., Gazdar G.,Klin E.,Pullum G. Generalized
 1985 Phrase Structure Grammar.- Oxford: Black-
 wells, 1985.
- Geometrix,1982 Geometrix Product Models // CAD/CAM Software,
 Information Rep.,1982.
- Gershma,1982 Gershma A. Building a Geological Expert Sy-

- stem for Dupmeter Interpretation // Proc. European CAI.-1982.-P.139-140.
- Goldberg et. al., 1982 Goldberg A.,Robson D.,Ingalls. Smalltalk-80: The Language and Its Implementation // Massa Chusetts: Addison-Welsey,1982.
- GRAFIXI,1985 GRAFIXI a Cad-Cam System. Рекламная информация фирмы GIXI.-1985.
- Grander,1980 Grander R. When Expectation Fails-Towards a Self-Correcting Inference System // Proc. Annual CAI.-Stanford,1980.-P. 301-305.
- Gray,1984 Gray,Neila B. Applications of AI for Organical Chemistry: Analysis of C-13 Spectra // AI.-1984.-Vol. 22.
- Green et. al., 1986 Green P.D.,Wood A.R. A Representational Approach to Knowledge Based Acoustic-Phonetic Processing in Speech Recognition // Proc. ICASSP.-Tokyo,23 Apr. 1986.-P. 1205-1208.
- Greiner et al., 1980 Greiner R.,Lenat D.B. Details of RLL-1// Rep. HPP-80-23 Stanford University.-Stanford (Calif.),1980.
- Grieamer et al., 1984 Grieamer J.H.,Hong S.J. YES/MVS: a Continuous Real Time Expert System // Proc. AAAI.-1984.
- Grinberg et al., 1980 Grinberg,Milton. A Knowledge-Based Desing System for Digital Electronics // Proc. AAAI.-1980.-P. 283-285.
- Grosz, 1977 Grosz B.J. The Representation and Use of Focus in a System for Understanding Dialog // Proc. 5th IJCAI.-Cambridge,Boston, 1977.-P. 67-76.
- Grosz et al., 1982 Grosz B.J.,Haas N.,Hendrix G.G. vol.32.- Dialogic: a Core Natural-Language Processing System// Proc. 9th Int. Conf. Computational Linguistics,Prague,Czechoslovakia, 1982.
- Grosz et al., 1987 Grosz B.J.,Appelt D.E.,Martin P.A. TEAM: an Experiment in the Design of Transportable Natural-Language Interfaces // Artificial Intelligence, 1987.-P.173-243.
- Groundwater,1984 Groundwater E.H. A Demonstration of an Ocean Surveillance Information Fusion Expert System // Proc. SPIE, Int. Society Optical Engineering.-Bellingham (Wash.). -1984. -Vol. 485, Applications of AI.
- Guida et al., 1983 Guida,Giovanni,Tasso C. An Expert Intermediary System for Interactive Document Retrieval // Automatica.-1983.-Vol. 19,N 6.-P.759-766.
- Gupta et al., 1984 Gupta N.K.,Seviora R.E. An Expert System Approach to Real Time System Debugging // Proc.

- CAI Applications, IEEE Computer Society.-1984.
- Hahn et al., 1980 Hahn W., Hoepfner W., Jameson A. The Anatomy of the Natural Language Dialogic System HAM-FRM // Natural Language Based Computer System.- Carl Hanser Verlag, 1980.-P. 347-354.
- Hajicova et al., 1981 Hajicova E., Sgall P. Towards Automatic Understanding of Technical Texts // Prague Bulletin Mathematical Linguistics.-1981.-N 36.-P. 5-19.
- Hajic, 1983 Hajic P. KODAS-a Natural Language Interface to a Simple Data Base // PBML.- 1983.- Vol. 39.-P. 65-76.
- Haley et al., 1983 Haley P., Kowalski J., McDermott J. PTRANS: a Rule-Based Management Assistant // Technical Report, Computer Science Dept., Carnegie-Mellon University.-Pittsburgh: Pa., Jan. 1983.
- Hammond, 1982 Hammond P. APES: a Detailed Description // Rep. 82/10, Computing Dept., Imperial College Science and Technology. 180 Queens Gate.-London: SW7 2BZ, UK, 1982.
- Hammond, 1983 Hammond P. APES: a User Manual // Rep. 82/9, Computing and Control Dept., Imperial College Science and Technology, 180 Queen's Gate.-London: SW7 2BZ, UK, May 1983.
- Hanson et al., 1978 Hanson A.-R., Riseman E.-M. VISIONS: a Computer System for Interpreting Scenes // Computer Vision Systems.-N.Y.: Academic Press, 1978.-P. 303-333.
- Harada, 1986 Harada T. NEC's Machine Translation System "PIVOT" // Japan Computer Quarterly.-1986.-N 64.-P. 24-31.
- Harandi et al., 1983 Harandi, Mehdi T. A General Purpose System for Inferencing // Proc. IBM University Study Conf., Raleigh N.C.-1983.-P. 131-139.
- Harandi et al., 1984 Harandi, Mehdi T. A Tree Based Knowledge Representation Scheme for Diagnostics Expert Systems // Proc. 1984 Conf. Intelligent Systems and Machines.-Rochester: Minn., 1984.
- Harris, 1977 Harris L.-R. User Oriented Data Base Query with the ROBOT Natural Language Query System // Int. J. Man-Machine Studies, 9.-1977.-P. 697-713.
- Havens et al., 1983 Havens W., Mackworth A. Representing Knowledge of the Visual World // IEEE Computer.-1983.-N 10.-P. 90-96.
- Hayes, 1984 Hayes Ph. J. Entity-Oriented Parsing // Internat Conf. Computational Linguistics. Proc. 22nd Annual Meeting of the Association for Computational Linguistics. Stanford University.

- July, 1984.-P. 212-217.
- Hayes et al., 1981a Hayes P.-J., Carbonel J.-G. Multy-Strategy Construction-Specific Parsing for Flexible Data Base Query and Update // Proc. 7th IJCAI.-Vancouver (Canada), 1981.-P. 432-439.
- Hayes et al., 1981b Hayes P.-J., Mouradian G.-V. Flexible Parsing // American J. Computational Linguistics.-1981.-Vol. 7(4).-P. 232-241.
- Hayes et al., 1983 Hayes P.-J., Reddy D.-R. Step Toward Graceful Interaction in Spoken and Written Man-Machine Communications // Int. J. Man-Machine Studies.-1983.-Vol. 19.-P. 231-284.
- Heiser et al., 1978 Heiser J.-F., Brooks R.-E., Ballard J.-P. Progress Report: a Computerized Psychopharmacology Advisor // Proc. 7th Colloquium Int. Neuro-Psychopharmacologicum.-Vienna (Austria), 1978.
- Helbig et al., 1983 Helbig H., Bottger H., String P. FAS-80 Ein Naturlisprachiges Auskunftssystem // VEB Roboten ZFT, WIB No 18.-1983.-P. 25-26.
- Hendrix, 1977 Hendrix G. LIFER: a Natural Language Interface Facility // SIGART Newsletter.-1977.-Vol. 61.-P. 25-26.
- Hewlett, 1986 Hewlett-Packard J. -1986, Nov.
- Hippe et al., 1987 Hippe Z., Kaczmarek A., Ampel T. Raport Z Badan' nad Komputerowym Systemem do Tlumaczenie Teksto'w Chemicznych Jezyka Rosyiskiego na Jezyk Polski.-Wyd. P.Rz.-Rzesro'w, 1987.
- Hoepfner et al., 1983 Hoepfner W., Christaler T., Morik K. Beyond Domains Independence: Experience with the Development of German Language Access System to Highly Diverse Background Systems // Proc. 8th IJCAI.-Karlsruhe, 1984.-P. 588-594.
- Hollan et al., 1984 Hollan J.-D., Hutchins E.-L., Weitzman L. STEAMER: an Interactive Inspectable Simulation Based Training System // AI Magazine.-1984.-Vol. 5, N 2.
- Hollander et al., 1983 Hollander C.-R., Iwasaki Y. The Drilling Advisor: an Expert System Application // Fundamentals of Knowledge Engineering, Teknowledge Report.-1983.
- Horstman, 1983 Horstman P.-W. Desing for Testability Using Logic Programming // Proc. Int. Test Conf., IEEE Computer Society Press, Silver Spring, Md.-1983.
- Hudson, 1981 Hudson D.-L. Rule-Based Computerization of Emergency Room Procedures Derived from Criteria Mapping // PhD Thesis, Computer Science Dept., Univer. California.-Los Angeles, 1981.
- Hudson et al., Hudson D.-L., Cohen M.-E. EMERGE, a Rule-Based

- 1984a Clinical Decision Making Aid // Proc. CAI Applications, IEEE Computer Society.- Dec. 1984.
- Hudson et al., 1984b Hudson D.L., Estrin T. Derivation of Rule-Based Knowledge from Established Medical Outlines // Computers in Biology and Medicine, Pergamon Press Ltd.-1984.-Vol.14,N 1.-P.3-13.
- Hudson et al., 1984a Hudson D.L., Estrin T. EMERGE, a Data-Driven Medical Decision Making Aid // IEEE Trans.-1984.-Vol. PAMI-6,N 1.
- Hutchins, 1986 Hutchins W.J. Machine Translation: Past, Present, Future.-New York, 1986.-P. 382.
- Ibuki et al., 1983 Ibuki J., Kume M., Nagao M. Japanese-to-English Title Translation System, TITRAN.-Its Outline and the Handling of Special Expressions in Titles // J. Information Processing.-1983.-Vol. 6, N 4.-P. 231-238.
- Isabelle et al., 1985 Isabelle P., Bourbeau L. TAUM-AVIATION: Its Technical Features and Some Experimental Results // Computational Linguistics.-1985.-Vol. 11,N 1.-P. 18-27.
- Ishizuka et al., 1981 Ishizuka M., Fu K.S., Yao J.T.P. SPERIL I: Computer-Based Structural Damage Assessment System // Rep. CE-STR-81-36, School Civil Engineering, Purdue Univer.-1981.
- IC MT, 1988 International Conference: New Directions in Machine Translation.-Budapest, 1988.
- Iwasaki et al., 1982a Iwasaki Y., Friesland P. SPEX: a Second-Generation Experiment Design System // Proc. AAAI.-1982.
- Iwasaki et al., 1982b Iwasaki, Yumi. SPEX: Skeletal Planner for Experiments // Rep. HPP-82-22, Computer Science Dept., Stanford Univer.-Stanford, (Calif.), Sept. 1982.
- Jacobs, 1983 Jacobs P.S. Generation in a Natural Language Interface // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 610-612.
- Johnson et al., 1981 Johnson P.E., Duran, A.S., Hassebrock, F. Expertise and Error Diagnostic Reasoning // Cognitive Science.-1981.-Vol. 5.-P. 235-283.
- Johnson et al., 1985 Johnson R., King M., des Tombe L. EUROTRA: a Multilingual System Under Development // Computational Linguistics.-1985.-Vol. 11,N 2-3.
- Kahn, 1984 Kahn G. On When Diagnostic Systems Want to do without Causal Knowledge // Proc. Advances AI, ECAI.-1984.
- Kahn et al., 1984a Kahn G., McDermott J. MUD, a Drilling Fluids Consultant // Technical Report, Department Computer Science, Carnegie-Mellon Univer.-

- Pittsburgh: Pa.,1984.
- Kahn et al., 1984b Kahn G.,McDermott J. The MUD system // Proc. CAI Application,IEEE Computer Society.- Dec. 1984.
- Kahn et al., 1984a Kahn G.,Nowlan S.,McDermott J. A Foundation for Knowledge Acquisition // Proc. IEEE Workshop Principles of Knowledge-Based Systems, IEEE Computer Society, Society Press, 1109 Spring Street, Silver Spring, Md.-1984.
- Kaihara et al., 1978 Kaihara S.,Koyama T.,Minamikawa T. A Rule-Based Physicians' Consultation System for Cardiovascular Diseases // Proc. Int. Conf. on Cybernetics and Society, IEEE.-Nov.,1978. -P. 85-88.
- Kaihara et al., 1982 Kaihara S.,Koyama T. Medical Consultation System with Practical Requirements-Development of MECS-AI // Computer Science and Technologies, Japan Annual Review in Electronic, Computers and Telecommunications.-Amsterdam (North-Holland),1982.-Vol. 4.
- Kaplan et al., 1979 Kaplan S.-J.,Mays E.,Joshi A.-K. A Technique for Managing the Lexicon in Natural Language Interface to a Changing Data Base // Proc. 6th IJCAI.-Tokio,1979.-P. 463-465.
- Kastner et al., 1982 Kastner J.-K.,Weiss S.-M.,Kulikowski C.-A. Treatment Selection and Explanation in Expert Medical Consultation: Application to a Model of Ocular Herpes Simplex // Proc. IEEE Conf. Medical Computer Science/Computational Medicine -Sept.,1982.-P. 420-427.
- Kastner et al., 1983 Kastner J.-K.,Weiss S.-M.,Kulikowski C.-A. An Efficient Scheme for Time-Dependent Consultation System // Proc. MEDINFO 4th World Conf. Medical Informatics.-North-Holland, 1983.-P. 619-622.
- Kastner et al., 1984 Kastner J.-K.,Chandler R.-D.,Weiss S.-M. An Expert Consultation System for Frontline Health Workers in Primary Eye Care // J. Medical Systems.-1984.-Vol. 8,N 5.
- Kay,1979 Kay M. Functional Grammar // Proc. 5th Annual Meeting of Berkeley Linguistic Society.-1979.
- Kedar-Cabelli 1984 Kedar-Cabelli. Analogy with Purpose in Legal Reasoning from Precedents // Rep. LPR-TR-17, Laboratory for Computer Science Research, Rutgers Univer.-July 1984.
- Kehler et al., 1984 Kehler T.-P.,Clemenson G.-D. An Application Development System for Expert Systems // Systems and Software.- 1984.-Vol. 3,N 1.
- Kelly et al., Kelly V.-E.,Steinberg L.-I. The CRITTER System:

- 1982 Analyzing Digital Circuits by Propagating Behaviors and Specifications // Proc. AAAI.- 1982.-P. 284-289.
- Kelly, 1984 Kelly V.E. The CRITTER System: Automated Critiquing of Digital Circuit Designs // Rep. LCSR-TR-55, Laboratory for Computer Science Research, Rutgers Univer.-May 1984.
- Kim et al., 1984 Kim J.H., Payton D.W., Olin K.E. A Context Dependent Automatic Target Recognition System // Proc. SPIE, Int. Society Optical Engineering, Bellingham, Wash., Applications AI.-1984.-Vol. 485.
- Kiremidjian et al., 1983 Kiremidjian G., Clarkson A., Lenat D. Expert System for Tactical Indications and Warning (I&W) Analysis // Proc. Army Conf. Application of AI to Battlefield Information Management, Rep. AD-A139 685, Battelle Columbus Laboratories.-Washington: D.C., 1983.
- Kirschner, 1982 Kirschner Z. A Dependency-Based Analysis of English for Purpose of Machine Translation // Эксплицитное описание языка и автоматическая обработка текстов.- Прага, 1982.-157 p.
- Kiselevich, 1983 Kiselevich S. TIMM-the Intelligent Machine Model // General Research Corporation Report, 5383 Holister Ave.-Santa Barbara (Calif.), Sept. 1983.
- Klahr et al., 1982 Klahr P., McArthur D., Narian S. SWIRL: Simulating Warfare in the ROSS Language // Rep. N-1885-AF, Rand Corporation.-September 1982.
- Klahr et al., 1984 Klahr P., Ellis J., Giarla W. TWIRL: Tactical Warfare in the ROSS Language // Rep. R-3518 -AF, Rand Corporation.-Santa Monica (Calif.), Oct. 1984.
- Klatt, 1983 Klatt D.H. The Problems of Variability in Speech Recognition and in Models of Speech Perception // Proc. 10th. Int. Congress of Phonetic Sciences Symposia.-Utrecht, 1983.-P. 287-298.
- Klinger, 1986 Klinger A. A Microcomputer-Based System for Cell Image Analysis // Proc. 8th IC on Pattern Recognition, Paris, IEEE.-1986.-P. 1316-1319.
- Koch et al., 1982 Koch D., Heicking W. SYSAN-Linguistischer Prozessor für FAS-80 // ZKI-Bericht.-Berlin, 1982.-P. 86.
- Kornell, 1984a Kornell Jim. A VAX Tuning Expert Built Using Automated Knowledge Acquisition // Proc. CAI Applications, IEEE Computer Society.-Dec. 1984.
- Kornell, 1984b Kornell Jim. Embedded Knowledge Acquisition to

- Simplify Expert Systems Development // Applied AI Reporter.- Aug.-Sept. 1984.-P.28-30.
- Kowalski et al., 1983 Kowalski T., Thomas D. The VLSI Design Automation Assistant: Prototype System // Proc. 20th Design Automation Conf., ACM and IEEE.- June 1983.
- Kudo et al., 1986 Kudo I., Nomura H. Lexical-Functional Transfer: a Transfer Framework in a Machine Translation System Based on LFG // COLING. - 1986.-P. 112-114.
- Kulikowski, 1980 Kulicowski C.A. AI Methods and Systems for Medical Consultation // IEEE Trans. - 1980.-Vol. PAMI-2, N 5.-P. 464-476.
- Kulikowski et al., 1980 Kulicowski C.A., Weiss S.M., Galen R.S. On the Diagnostic Frontier // Diagnostic Medicine.- Nov.-Dec. 1980.
- Kunz et al., 1984 Kunz J.C., Kehler T.R., Williams M.D. Applications Development Using a Hybrid AI Development System // AI Magazine.- Fall 1984.- Vol. 5, N 3.
- Kurzweil, 1985 Kurzweil introduced word recognition peripheral // Computer Design.- 1984. -June. -P.69-76.
- Kushima et al., 1986 Kushima H., Uchida H. Fujitsu's Machine Translation Systems // Japan Computer Quarterly.- 1986.-N 64.-P. 19-23.
- Laffey et al., 1984a Laffey T.J., Perkins W.A., Nguyen T.A. Reasoning about Fault Diagnosis With LES // Proc. CAI Applications, IEEE Computer Society.- Dec. 1984.
- Laffey et al., 1984b Laffey T.J., Perkins W.A., Firschein O. LES: a Model-Based Expert System for Electronic Maintenance // Proc. Joint Services Workshop on AI in Maintenance.- 4-6 Oct. 1984.-P. 1-17.
- Lebowitz, 1985 Lebowitz M. RESEARCHER: an Experimental Intelligent Information System // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 858-862.
- Lesmo et al., 1985 Lesmo L., Torasso P. Weighted Interaction of Syntax and Semantics in Natural Language Analysis // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 772-778.
- Levine et al., 1981 Levine M.D., Shaheen S.I. A Modular Computer Vision System for Picture Segmentation and Interpretation // IEEE Trans. -1981.-Vol. PAMI-3, N 5.-P. 540-556.
- Lewis et al., 1983 Lewis J.W., Lynch F.S. GETREE: a Knowledge Management Tool // Proc. Trends and Applications Conf. Automating Intelligent Behavior, IEEE Press.-May 1983
- Lindberg et al., Lindberg D.A.B., Sharp G.C., Kingsland L.C.

- 1980 Computer Based Rheumatology Consultant // Proc. 3rd World Conf. Medical Informatics. -1980.-P. 1311-1315.
- Lobanov,1988 Lobanov B. The Phonemophone Text-to-Speech System // Proc. 11th ICPHS.-Tallin,1988.-P. 123-129.
- Lougheed et. al., 1980 Lougheed R.M.,McCurbey D.L. The Cytocomputer: a Practical Pipelined Image Processor // Proc. 7th Ann. Symp. Computer Architecture, IEEE.-1980.-P. 271-277.
- Lowe,1985 Lowe D.G. Visual Recognition from Spatial Correspondence and Perceptual Organization // Proc. 9th IJCAI.-Los Angeles,18-23 Aug. 1985.-P. 953-959.
- Luckhart et al., 1983 Luckhart H-D.,Maas H-D. SUSY - Handbuch fur Transfert und Synthese // Linguistische Arbeiten des SB-100;Neue Folge.-Saarbrücken, 1983.-P. 178.
- Luria,1982 Luria M. Dividing Up the Gestion Answering Process // Proc. National CAI (AAAI-82).-Pittsburg,1982.-P. 71-74.
- Lytle,1980 Lytle E.G. Junction Grammar: Theory and Application // 6th LACUS Forum.-Columbia (S.Carolina),1980.-P. 305-343.
- Mann,1983 Mann W.C. An Overview of the Penman text Generation System // Proc. Nat. CAI. -Washington, 1983.-P. 261-265.
- Marsh,1984a Marsh A.K. NASA to Demonstrate AI in Flight Operations // Aviation Week Space Technology.-Sept. 17,1984.
- Marsh,1984b Marsh A.K. C. AI Laboratory // Ford Aerospace Communications Corporation.-Houston (Texas): Private Communication,1984.
- Marsh,1984c Marsh A.K. Pace of AI Research Shows Acceleration // Aviation Week Space Technology.-1 Dec. 1983.
- Martin,1981 Martin P. An Application Development without Programmers,Savant Institute.-1981.-P. 327.
- Martin et al., 1971 Martin W.A.,Fateman R-J. The MACSYMA System // Proc. 2nd Symp. Symbolic and Algebraic Manipulation.-March 1971.-P.59-75.
- Martin et al., 1983 Martin P.,Appelt D.,Pereira F. Transportability and Generality in a Natural Language Interface System // Proc. 8th IJCAI.-Karlsruhe, 1983.-P. 573-581.
- Masui et al., 1983 Masui S.,McDermott J.,Sobel A. Decision-Making in Time-Critical Situations // Proc. 8th IJCAI. -1983. -P. 233-235.
- Matsuyama et al., Matsuyama T.,Hwang V. SIGMA: a Framework for

- 1985 Image Understanding. Integration of Bottom-up and Topdown Analysis // Proc. 9th IJCAI.- Los Angeles.- 1985.-P. 908-915.
- McCune et al., 1983a McCune B.P., Drazovich R.J. Radar with Sight and Knowledge // Defense Electronics.- Aug. 1983.
- McCune et al., 1983b McCune B.P., Tong R.M., Dean J.S. RUBRIC: a System for Rule-Based Information Retrieval // Proc. 7th Int. Computer Software and Applications Conf. (COMPSAC-83), IEEE Computer Society.-Nov. 1983.-P. 166-172.
- McDermott, 1980 McDermott J. KI: an Expert in the Computer Systems Domain // Proc. AAAI.-1980.
- McDermott, 1981a McDermott J. RI's Formative Years // AI Magazine.-1981.-Vol. 2.-N 2.
- McDermott, 1981b McDermott J. Domain Knowledge and the Design Process // ACM/IEEE 13th Design Automation Conf. Proc.-1981.
- McDermott, 1982a McDermott J. XSEL: a Computer Salesperson's Assistant // Machine Intelligence, 10.-1982.-P. 325-337.
- McDermott, 1982b McDermott J. RI: a Rule-Based Configurer of Computer Systems // AI.- 1982.-Vol. 19, N 1.
- McDermott, 1982c McDermott J. Domain Knowledge and the Design Process // Design Studies.- 1982.-Vol. 3.-N 1.
- McDermott, 1984 McDermott J. Building Expert Systems.// AI Applications Business, Norwood N.J.-Ablex, 1984.
- McDonald et al., 1985 McDonald D.D., Pustejovsky J.D. Description-Directed Natural Language Generation // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 799-805.
- McKeown, 1983 McKeown K.R. Focus Constraints on Language Generation // Proc. 8th IJCAI.-Karlsruhe, 1983.-P. 582-587.
- McKeown et al., 1985 McKeown K.R., Wish M., Muthews K. Tailoring Explanation for the User // Proc. 9th IJCAI.- Los Altos (Calif.), 1985.-P. 795-798.
- McKeown, 1982 McKeown D.M. Concept Maps // Proc. DARPA Image Understanding Workshop.-Sept. 1982.-P. 142-153.
- McKeown et al., 1983 McKeown D.M., McDermott J. Toward Expert Systems for Photo Interpretation // IEEE Proc. Trends and Applications.-1983.-P. 33-39.
- McKeown et al., 1984 McKeown D.M., Harvey W.A., McDermott J. Rule-Based Interpretation of Aerial Imagery // Proc. IEEE Workshop on Principles of Knowledge-Based Systems, IEEE Computer Society Press, 1109 Spring Street, Silver Spring, Md.-1984.
- McWilliams, 1986a McWilliams J. International Symposium on Machine Translation // Japan Computer Quart-

- erly.-1986.-N 64.-P. 4-11.
- McWilliams,1986 J. Machine Translation Systems at EXPO'85 // Japan Computer Quarterly.-1986.-N64.-P. 55-58.
- Melby,1982 Melby A.K. Multi-Level Translation Aids in a Distributed System // Proc. 9th Int. Conf. Computational Linguistics. -Prague: Academia, 1982.-P.215-228.
- Meldman,1975 Meldman J.A. A Preliminary Study in Computer-Aided Legal Analysis // Phd Thesis Departments Electrical Engineering and Computer Science,MIT.-Aug. 1975.
- Michaelsen,1982 Michaelsen R. A Knowledge-Based System for Individual Income and Transfer Tax Planning // PhD Thesis,University Illinois,Accountin Dept.,Campaign-Urbana.-1982.
- Michaelsen et al., 1983 Michaelsen R.,Michie D. Expert Systems In Business // Dalamat.-Nov. 1983.-P.240-246.
- Michalski et al., 1980a Michalski R-S.,Chilausky R-L. Learning by Being Told and Learning from Examples: an Experimental Comparison of Two Methods of Knowledge Acquisition in the Context of Developing an Expert System for Soybean Disease Diagnosis. A Special Issue on Knowledge Acquisition and Induction // Policy Analysis and Information Systems.-1980.-N 2.
- Michalski et al., 1980b Michalski R-S.,Chilausky R-L. Knowledge Acquisition by Encoding Expert Rules Versus Computer Induction from Examples: a Case Study Involving Soybean Pathology // Int. J. Man-Machine Studies.-1980.-Vol. 12.-P. 63-87.
- Michalski et al., 1982 Michalski R-S.,Davis J.H.,Bisht V.S. PLANT/ ds: an Expert Consulting System for the Diagnosis of Soybean Diseases // Proc. 5th European CAI.-Orsay(France),July 1982.
- Michalski et al., 1983 Michalski R-S.,Baskin A.B. Integrating Multiple Knowledge Representations and Learning Capabilities in an Expert System the ADVISE System // Proc. 9th IJCAI.-1983.-P. 256-258.
- Michie et al., 1984 Michie D.,Muggleton S.,Riese C. RULEMASTER: A Second-Generation Knowledge-Engineering Facility // Proc. CAI Applications,IEEE Computer Society.-Dec. 1984.
- Mikulich,1982 Mikulich L-L. Natural Language Dialogue Systems: a Pragmatic Approach // Machine Intelligence.-1982.-P. 383-396.
- Miller,1983 Miller P-L. Medical Plan-Analysis: the ATTENDING System // Proc. IJCAL. -1983.-P. 239-241.

- Miller et al., 1984 Miller P.-L., Black H.-R. Medital Plan-Analysis by Computer Critiquing the Pharmacologic Management of Essential Hypertension // Computers and Biomedical Research.-1984.-Vol. 17.-P. 38-54.
- Minton et al., 1985 Minton S., Hayes P.-J., Fain J. Controlling Search in Flexible Parsing // Proc. 9th IJCAI.-Los Altos (Calif.), 1985.-P. 785-787.
- Mitchell et al., 1983 Mitchell T.-M., Steinberg L.-I., Kedar-Cabelli S. An Intelligent Aid for Circuit Redesign // Proc. AAAI.-1983.-P. 274-278.
- Mizoguchi, 1983 Mizoguchi F. PROLOG Based Expert System // New Generation Computing, Japan: Ohmsha Ltd. and Springer-Verlag.-1983.-Vol. 1.
- Moore et al., 1986 Moore G.-W. et al. Multilingual Translation Techniques in the Analysis of Narrative Medical Text // Computer Methods and Programs in Biomedicine.-1986.-Vol. 22.-P. 35-42.
- Moore et al., 1984 Moore R.-L., Hawkinson L.-B., Knickerbocker C.-G. A Real-Time Expert System for Process Control // Proc. CAI Applications, IEEE Computer Society.-Dec. 1984.
- Morris, 1984 Morris D. LISP Shows First Process-Control Expert System // Electronic Engineering Times.-13 Aug. 1984.
- Moser, 1984 Moser M.-G. Domain Dependent Semantic Acquisition // Proc. 11th CAI Applications.-Denver, 1984.-P. 13-17.
- MT Summit, 1987 Machine Translation Summit. Manuscripts and Program.-Hakone, 1987
- Mueckstein, 1983 Mueckstein E.-M. Q-TRANS: Query Translation into English // Proc. 8th IJCAI.-Karlsruhe, 1983.-P. 624-626.
- Mulder, 1985 Mulder J.-A. Using Discrimination Graphs to Represent Visual Interpretations that are Hypothetical and Ambiguous // Proc. 9th IJCAI.-Los Angeles, 1985.-P. 905-907.
- Mulsant et al., 1984 Mulsant B., Servan-Schreiber D. Knowledge Engineering: a Daily Activity on a Hospital Ward // Computers and Biomedical Research.-1984.-Vol. 17.-P. 71-91.
- Nagao et al., 1980 Nagao M., Matsuyama T. A Structural Analysis of Complex Aerial Photographs.- New York: Plenum, 1980.
- Newell, 1969 Newell A. Heuristic Programming: Ill Structured Problems // Progress in Operation Research. Vol. 3. - New York: Wiley and Sons, 1969.
- Niemann et al., 1984 Niemann H., Bunke H., Hofmann I. Diagnostic Inferences from Image Sequences - a Knowledge-

- Based Approach // Proc. CAI Applications, IEEE Computer Society.-Dec.1984.
- Nii et al., 1979 Nii H., Aiello N. AGE (Attempt to Generalize): a Knowledge-Based Program for Building Knowledge-Based Programs // Proc. IJCAI.-1979. -P. 209-219.
- Nirenburg et al., 1986 Nirenburg S., Raskin V., Tucker A. On Knowledge-Based Machine Translation // COLING.-1986. -P. 627-632.
- Nirenburg, 1987 Nirenburg S., ed. Machine translation: theoretical and methodological issues.-Cambridge (Mass., USA), 1987
- Nishida et al., 1986 Nishida F., Fujita Y., Takamatsu Sh. Construction of a Modular and Portable Translation System // COLING.-1986.-P. 649-651.
- Niyogi et al., 1986 Niyogi D., Srihari S. A Rule-Based System for Document Understanding // Proc. 5th Nat. Conf. AAAI.-Philadelphia, 11-15 Aug. 1986. -P. 789-793.
- Nomura et al., 1986 Nomura H., Naito Sh., Katagiri Y. Translation by Understanding: a Machine Translation System LUTE // COLING. -1986.-P. 621-626.
- Norman et al., 1975 Norman D.A., Rumelhart D.E., Explorations in Cognition.-San Francisco: Freeman, 1975.
- Norvig, 1983 Norvig P. Frame Activated Inference in a Story Understanding Program // Proc. 8th IJCAI.-Karlsruhe, 1983.-P. 624-626.
- O'Connor, 1984 O'Connor D.E. Using Expert Systems to Manage Change and Complexity in Manufacturing. // AI Applications for Business, Norwood N.J.-Ablex, 1984.
- Ogawa et al., 1984a Ogawa H., Fu K.S., Yao J.T.P. An Expert System for Damage Assessment of Existing Structures // Proc. Conf. AI Applications. IEEE Computer Society.-Dec. 1984.
- Ogawa et al., 1984b Ogawa H., Fu K.S., Yao J.T.P. Knowledge Representation and Inference Control of SPERIL-2. ACM // Conf. Proc.-Nov. 1984.
- Ogonovski, 1987 Ogonovski A. MENTAT: an Intelligent and Cooperative Natural Language DB Interface // Proc. 7th Int. Workshop Expert Systems and Their Applications.-Avignon, 1987.-P. 1271-1283.
- Orciuch et al., 1984 Orciuch E., Frost J. ISA: an Intelligent Scheduling Assistant // Proc. Conf. AI Applications, IEEE Computer Society.-Dec.1984.
- Papagailij, 1986 Papagailij B.C. Word Expert Semantics. An Interlingual Knowledge-Based Approach // Distributed Language Translation / Ed. Toon

- Paris,1985 Witkam. -Dortrecht: Reverton,1986.-P. 254.
Paris C.L. Description Strategies for Naive
and Expert Users // Proc. 23th Annual Meet-
Assoc. Comput. Linguistics.-Chicago,1985.-P.
238-245.
- Parker et al., Parker R.E.,Kiselewich S.J. The Modeling of
1984 Human Cognitive Decision Processes in the In-
telligent Machine Model // Report,AI Labora-
tory,General Research Corporation.-Santa Bar-
bara (Calif.),1984.
- Patel-Schneider, Patel-Schneider P. Small Can Be Beautiful in
1984 Knowledge Representation // Proc. IEEE
Workshop in Principles of Knowledge-Based Sys-
tems,IEEE Computer Society Press,Silver
Spring,Md.-1984.
- Paterson et al., Paterson A.,Niblett T. ACLS User Manual // Re-
1982 port,Oxford,Canal Street,15,Intelligent Ter-
minals,UK 082 6BH.-1982.
- Paxton,1978 Paxton W.H. The Language Definition System
// Understanding Spoken Language. - New-York:
Elsevier-North Holland.-1978.-P. 17-40.
- Pazzani,1983 Pazzani M.J. Interactive Script Instantiation
// Proc. National CAI.-Washington,1983.-
P. 320-326.
- Pazzani et al., Pazzani M.,Engelman C. Knowledge Based Ques-
1983 tion Answering // Proc. Conf. Applied Natural
Language Processing.-Santa Monica,1983.
- Perrone,1983 Perrone J. Expert Systems Get Personal: Mo-
deling Expertise with the IBM PC // Report,
Jeffery Perronc Associates,Inc.,3685 17th
Street.-San Francisco (Calif.),1983.
- Petric,1981 Petric S. Field-testing the Transformational
Question Answering (TQA) System // Proc.
19th Annual Conf. ACL.-1981.-P. 35-36.
- Pisano et al., Pisano A.D.,Jones H.L. An Expert Systems Ap-
1984 proach to Adaptive Tactical Navigation //
Proc. CAI Applications,IEEE Computer Society.
-Dec. 1984.
- Pollack et al., Pollack J.J.,Zamora A. Automatic Spelling
1984 Correction in Scientific and Scolary Text //
CACM.-1984.-Vol. 27.-P. 358-368.
- Popp et al., Popp W.G.,Schlink B. Judith,a Computer Pro-
1975 gram to Advise Lawyers in Reasoning a Case
// Jurimetrics J.-Summer 1975.-Vol. 15.
-P. 303-314.
- Puppe et al., Puppe F.,Bernhard. Overview on MED1: a Heu-
1983 ristic Diagnosis System with an Efficient
Control Structure // Rep. SEKI-83-02,Fach-
bereich Informatik,Univer. Kaiserslautern.

- West Germany,1983.
- Ramo et al.,
1986 Ramo E.G.,Salminen J. An Experimental Study of Low Cost Image Workstation Architectures // Proc. 8th Int. Conf.Pattern Recognition, IEEE Computer Society Press.-1986.-P. 34-38.
- Rauch,1984 Rauch H-E. Expert Systems as Automated Decision Aids // Proc. ICASSP.-Wash.,1984.-Vol. 2.-P. 1-4.
- Reboh et al.,
1982 Reboh R.,Reiter J.,Gaschnig J. Development of a Knowledge-Based Interface to a Hydrological Simulation Program // SRI Technical Report 3477,SRI Int.-Menlo Park (Calif.), May 1982.
- Reggia,1978a Reggia J-A. Representing and Using Medical Knowledge for the Neurological Localization Problem // Rep. TR-693,Computer Science Dept.,Univer. Maryland.-Sept. 1978.
- Reggia,1978b Reggia J-A. A Production Rule-System for Neurological Localization // Proc. 2nd Annual Symp. on Computer Applications in Medical Care,IEEE.-Nov. 1978.
- Reggia,1981 Reggia J-A. Knowledge-Based Decision Support Systems: Development Through KMS // Rep. TR-1121 Computer Science Dept.,Univer. Maryland.-College Park: Md,1981.
- Reggia et al.,
1983 Reggia J-A.,Nau D-S.,Wang P-Y. Diagnostic Expert Systems Based on a Set Covering Model // Int. J. Machine Studies.-1983.-Vol. 19. P. 437-460.
- Riese,1985 Riese C. Control Strategies in RULEMASTER // Rep. R1-RS-00296,Radian Corporation,8501 Mo-Pac Blvd.Austin,Tex.-Jan. 1985.
- Riseman et al.,
1977 Riseman E-M.,Hanson A-R. A Contextual Post-processing System for Error Correction Using Binary n-Grams // IEEE Trans.-1977.-Vol.Comp.-23,N 5.-P. 480-493.
- Roach et al.,
1984 Roach J.,Lee S.,Wilcke J. An Expert System that Criticizes Decisions in Combination Drug Therapy // Proc. CAI Applications,IEEE Computer Society.-Dec. 1984.
- Roach et al.,
1985 Roach J-W.,Virkar R-S. POMME: a Computer-Based Consultation System for Apple Orchard Management Using PROLOG // Technical Report, Department of Computer Science and Applications,Virginia Polytechnic Institute and State University.-1985.
- Robinson,1982 Robinson J.J. Diagram: a Grammar for Dialogues // CACM.-1982.-25(1).-P. 27-42.
- Rodewald,1984 Rodewald L.E. BABY: an Expert Systems for Patient Monitoring in a Newborn Intensive

- Care Init // M.S. Thesis Computer Science Dept.,Univer. Illinois,Champaign-Urbana--1984.
- Rosenbloom et al., 1984 Rosenbloom P.S.,Laird J.E.,Dermott J. RI-SOAR, an Experiment in Knowledge-Intensive Programming in a Problem-Solving Architecture // Proc. IEEE Workshop on Principles of Knowledge-Based Systems,IEEE Computer Society, Press, 1109 Spring Street,Silver Spring,Md.-1984.
- Rosenthal et al., 1984 Rosenthal D.A.,Bajacsy R. Visual and Conceptual Hierarchy: a Paradigm for Studies of Automated Generation of Recognition Strategies // IEEE Trans.-1984.-Vol. PAMI-6, N 3.-P. 319-325.
- Rosner,1986 Rosner D. When Marico Talks to Siegfried-Experience from a Japanese/German Machine Translation Project // COLING.-1986.-P. 652-654.
- Ross et al., 1984 Ross J.,MacAllister. Practical and Continuous Speech Recognition // Computer Design.-June 1984.-P. 69-76.
- Rummel,1986 Rummel P. GSS-A Fast Model-Based Ctrey Scale Sensor System for Workpiece Recognition // Proc. 6th Int. Conf. Pattern Recognition.-Paris, 1986.-P. 18-21.
- Saluveer et al., 1986 Saluveer M.,Oim H. Rules and Reasoning in Text Comprehension // Neue Ansätze in Maschineller Sprachübersetzung: Wissenrepräsentationen und Textbezug.-Tübingen: Niemeyer, 1986.-P. 139-163.
- Sanamrad et al., 1986 Sanamrad M.A.,Matsumoto H. PERSIS: a Natural Language Analyzer for Persian // J. Information Processing.-1986.-P. 271-279.
- Scarl,1984 Scarl E. Applications of a Non-Role Knowledge-Based System to NASA Scheduling and Monitoring Problems // IECEC Proc.- San-Francisco,Aug. 1984.
- Scarl et al., 1984 Scarl E.,Jamieson J.,Dehrne C. Knowledge-Based Fault Monitoring and Diagnosis in Space Shuttle Propellant Loading // Proc. Nat. Aeronautics and Electronics Conf.,Dayton.-Ohio,1984.
- Schmolze et al., 1982 Schmolze J.G.,Brachman R.J.(eds.) // Proc. 1981 KL-ONE Workshop,Fairchild Laboratory for AI Research.-Palo Alto (Calif.),May 1982.
- Schmolze et al., 1983 Schmolze J.G.,Lipkis T.A. Classification in the KL-ONE Knowledge Representation System // Proc. IJCAL. -1983.-P. 330-332.
- Schor,1984 Schor M.I. Using Declarative Knowledge Rep-

- representation Techniques: Implementing Truth Maintenance in OPS5 // Proc. CAI Applications, IEEE Computer Society.-Dec.1984.
- Schubin et al., 1982 Schubin H.,Ulrich J.W. IDT: an Intelligent Diagnostic Tool // Proc.AAAI.-1982.-P.290-295.
- Schwabe et al., 1982 Schwabe W.,Jamison L.M. A Rule-Based Policu-Level Model of Nonsuperpower Behavior in Strategic Conflicts // Rep. R-2962-DNA,Rand Corporation.-Dec. 1982.
- Sembugamoorthy,1979 Sembugamoorthy V. PLAS,a Paradigmatic Language Aquisition System: an Overview // Proc. 6th IJCAI.-Tokyo,1979.-P. 788-790.
- Shank et al., 1977 Shank R.C.,Abelson R. Scripts,Plans,Goals and Understanding.-New York: Lawrence Erlbaum Press,1977.
- Shank et al., 1979 Shank R.S.,Kolodner J.L. Retrieving Information from an Episodic Memory or Why Computer's Memories Should Be More Like People's // Proc. 6th IJCAI.-Tokyo,1979.-P. 766-768.
- Shimizu et al., 1983 Shimizu T.,Sakamura K. MIXER: an Expert System for Microprogramming // Proc. 16th Annual Microprogramming Workshop,ACM.-Oct.1983.-P. 168-175.
- Shipman et al., 1979 Shipman D.W.,Marcus M.P. Towards Minimal Data Structures for Deterministic Parsing // Proc. 6th IJCAI.-Tokyo,1979.-P. 815-817.
- Shortliffe,1976 Shortliffe E.H. Computer Based Medical Consultations: MYCIN .- New York: American Elsevier,1976.
- Shortliffe et al., 1981 Shortliffe E.H.,Scott A.C.,Bischoff M.B. ONCOCIN: an Expert System for Oncology Protocol Management // Proc. 7th IJCAI.-1981.-P. 876-881.
- Shortliffe et al., 1984 Shortliffe E.H.,Scott A.C.,Bischoff M.B. An Expert System for Oncology Protocol Management // Rule-Based Expert Systems.-Addison-Wesley,1984.
- Shreve et al., 1986 Shreve G.M.,Gommlich K.,Bohm E. Pseudocode Templates for a Microcomputer-Based Computer Assisted Translation System // Linguistic Arbeits-Berichte.-Leipzig: Karl-Marx Univ., 1986.-N 5-6.-P. 38-56.
- Simmonds,1985 Simmonds B. Elsin Instrumentation // Control and Instrumentation.-Sept.1985.
- Simmons et al., 1972 Simmons R.,Slocum J. Generating English Discourse from Semantic Networks // CACM.-1972.-Vol. 15,N 10.-P. 891-905.
- Slagle et al., 1983 Slagle J.,Gaynor M. Expert System Consultation Control Strategy // Proc. AAAI.-1983.-

- P. 369-372.
- Slocum,1985 Slocum J. A Survey of Machine Translation: Its History,Current Status,and Future Prospects // Computational Linguistics.-1985.-Vol. 11,N 1.
- Slocum et al., 1984 Slocum J.,Bennet W.S.,Bear J. METAL: the LRC Machine Translation System.- Austin (Texas): Univ. Texas,1984.
- Smith et al., 1980 Smith D.E.,Clayton J.E. A Frame-Based Production System Architecture // Proc. AAAI.-1980.-P. 154-156.
- Smith,1983 Smith R.G. STROBE: Support for Structured Object Knowledge Representation // Proc.8th IJCAI.-1983.-P. 855-858.
- Smith et al., 1984a Smith D.E.,Clayton J.E. Another Look At Frames. // Rule-Based Expert Systems.-Addison - Wesley,1984.-P. 441-452.
- Smith et al., 1984b Smith J.W.,Speicher C.,Chandrasekaran B. Expert Systems as Aids for Interpretive Reporting // Proc. 7th Hawaii Int. Conf.System Sciences.-1984.
- Smith et al., 1984c Smith R.G.,Young R.L. The Design of the DIPMETER ADVISOR System // ACM Conf. Proc. - Nov. 1984.
- Stallman et al., 1979 Stallman R.M.,Sussman G.J. Problem Solving About Electrical Circuits. // AI,An MIT Perspective,MIT Press.-1979.-Vol. 1.-P. 30-91.
- Stansfield,1984 Stansfield S.A. ANGV: a Rule-Based Expert Systems for Identifying and Isolating Coronary Vessels in Dieital Angiograms // Proc. CAI Applications,IEEE Computer Society.-Dec. 1984.
- Stefik,1978 Stefik M. Inferring DNA Structures from Segmentation Data // AI.-1978.-Vol. 11.-P. 85-114.
- Stern et al., 1986 Stern P.E.,Eskenazi M.,Memmi D. An Expert Systems for Spuch Spectrogram Reading // Proc. ICASSP.-Tokyo,1986.-23 Jan.1984.-P. 1193-1196.
- Swartout,1977 Swartout W.R. A Digitatis Therapy Advisor wiht Explanations // MIT Technical Rep.TR-176.-Febr. 1977.
- Szpakowicz et al, 1982 Szpakowicz S.,Swifzinski M. A Simple Dialogue in Polish: Interactive Railway Guide // Computers and AI.-1982.-N 5.-P. 419-425.
- Tamura et al., 1984 Tamura H.,Yokoya N. Image Database Systems: a Survey // Pattern Recognition.-1984.-Vol. 17,N 1.-P. 29-43.
- Taniguchi et al., Taniguchi R.,Kawaguchi E. Knowledge-Based

- 1986 Image Analysis of LANDSAT Data // Proc. 6th Int. Conf. Pattern Recognition.-Paris, 1986. -P. 1236-1239.
- Tanimoto, 1982 Tanimoto S.L. Advances in Software Engineering and their Relations to Pattern Recognition and Image Processing // Pattern Recognition.-1982.-Vol. 15, N 3.-P. 113-120.
- Thompson et al., 1983 Thompson W.B., Johnson P.E., Moen J.B. Recognition-Based Diagnostic Reasoning // Proc. IJCAI.-1983-P. 236-2380.
- Tomita et al., 1986 Tomita M., Carbonell J.G. Another Stride Towards Knowledge-Based Machine Translation // COLING.-1986.-P. 633-638.
- Tong, 1986 Tong L. English-Malay Translation System: a Laboratory Prototype // COLING.-1986.-P. 639-642.
- Tsutsumi, 1986 Tsutsumi T. A Prototype English-Japanese Machine Translation System for Translating IBM Computer Manuals // COLING.-1986.-P. 646-648.
- Tucker, 1984 Tucker A.B. A Perspective on Machine Translation: Theory and Practice // CACM.-1984.-Vol. 27.-N 4.-P. 322-329.
- Tufis, 1982 Tufis D. SDRL: a Dialogue System for Romanian Language. Research Rep. AI-FDO22, CIMI.-Bucharest (Romania), Oct. 1982.-P. 12.
- Underwood, 1982 Underwood W.E. A CSA Model-Based Nuclear Power Plant Consultant // Proc. AAAI.-1982.-P. 302-305.
- Valtorta et al., 1984 Valtorta M.G., Smith B.T., Loveland D.W. The Graduate Course Advisor: a Multi-Phase Rule-Based Expert System // Proc. IEEE Workshop on Principles of Knowledge-Based Systems, IEEE Computer Society Press, 1109 Spring Street, Silver Spring, Md.-1984.
- Vauquois et al., 1985 Vauquois B., Boitet Ch. Automated Translation at Grenoble University // Computational Linguistics.-1985.-Vol. 11, N 1.-P. 28-36.
- Vesonder et al., 1983 Vesonder G.T., Stodfo S.J., Zielinski J.E. ACE: an Expert System for Telephone Cable Maintenance // Proc. 8th IJCAI.-1983.-P. 116-121.
- Walker et al., 1987 Walker T.C., Miller R.K. Expert Systems 1987. An Assessment of Technology and Applications // SEAI. Technical Publications Madison, GA,-1987.
- Walser et al., 1976 Walser R.L., McCormick B.H. Organization of Clinical Knowledge in MEDICO // Proc. 3rd Illinois Conf. Medical Information Systems, Champaign-Urbana, Univ. Illinois.-Nov. 1976.
- Walser et al., Walser R.L., McCormick B.H. A System for Prim-

- 1977 ing a Clinical Knowledge Base // Proc. AFIPS Nat. Computer Conf.-1977.-P. 301-307.
- Waltz,1978 Waltz D.L. An English Language Question Answering System for a Large Relational Database // CACM.-1978.-P. 526-539.
- Wang et al., 1986 Wang C.H.,Srihari S.N. Object Recognition in Structured and Random Enviroments: Locating Addresses Blocks on Mail Pieces // Proc. 5th National Conf. AAAI.-Philadelphia,11-15 Aug. 1986.-P. 1133-1137.
- Wasserman et al., 1983 Wasserman K.,Lebowitz M. Representing Complex Physical Objects // Cognition and Brain Theory.-1983.-Vol. 6,N 3.-P. 333-352.
- Waterman et al., 1980 Waterman D.A.,Peterson M.A. Rule-Based Models of Legal Expertise // Proc. Annual Nat. CAI.-1980.
- Waterman et al., 1981 Waterman D.A.,Peterson M.A. Models - Legal Decision-Making // Rep. R-2717-ICJ,Rand Corp.-1981.
- Waterman et al., 1983 Waterman D.A.,Hayes-Roth F. An Investigation of Tools for Building Expert Systems. //Building Expert Systems,Reading.-Mass.: Adison-Wesley,1983.
- Waterman,1986 Waterman D.A. A Guide to Expert Systems. - New York: Addison-Welse,1986.
- Weischedel et al., 1980 Weischedel R.M.,Black J. Responding to Potentially Unparsable Sentences // American J. Computational Linguistics.-1980.-N 6.-P. 97-109.
- Weiss et al., 1978 Weiss S.M.,Kulikowski C.A.,Amarel S. A Model-Based Method for Computer-Aided Medical Decision-Making // AI.-1978.-Vol. 11.-P. 145-172.
- Weiss et al., 1980 Weiss S.M.,Kern K.B.,Kulikowski C.A. A Guide to the Use of the EXPERT Consultation System // Technical Rep. CBM-TR-94,Computer Science Dept.,Rutgers Univ.-New Brunswick, Jan. 1980.
- Weiss et al., 1981a Weiss S.M.,Kulikowski C.A. Developing Microprocessor-Based Expert Models for Instrument Interpretation // Proc. 7th IJCAI.-1981.-P. 853-855.
- Weiss et al. 1981b Weiss S.M.,Kulikowski C.A. Expert Consultation Systems: the EXPERT and CABNET Projects // Machine Intelligence, Infotech State of the Art Report, Pergamon Infotech Ltd.-1981 - Ser. 9,N 3. -P. 339-353.
- Weston et al., 1987 Weston C.D.,Stewart G.A. Workstations // Byte -1987.-Vol. 12,N 2.-P. 85-97.
- Whitlock et al., Whitlock P.J.,Wood M.M.,Chandler B.J. Str-

- 1986 tegies for Interactive Machine Translation: the Experience and Applications of the UMIST Japanese Project // COLING.-1986.-P. 329-334.
- Wilensky,1979 Wilensky R. Understanding Complex Situations // Proc. 6th IJCAI.-Tokyo,1979.-P. 954-959.
- Wilensky,1981a Wilensky R. A Knowledge-Based Approach to Language Processing: a Progress Report // Proc. 7th IJCAI.-Vancouver,1981.-P. 25-30.
- Wilensky,1981b Wilensky R. Meta-Planning: Representing and Using Knowledge about Planning in Problem Solving and Natural Language Understanding // Cognitive Science.-1981.-Vol. 5,N 3.
- Wilensky,1982 Wilensky R. Talking to UNIX in English: an Overview of UC // Proc. Nat. CAI. -Pittsburgh 1982.-P. 103-106.
- Winn,1987 Winn L. Voice Recognition: Understanding the Masters Voice // PC Magazine, 1987.
- Winograd,1983 Winograd T. Language as a Cognitive Process. Vol. 1: Syntax.-Addison-Wesley Publishing Co. -1983.-P. 357-401.
- Wong et al., Wong C.M.,Crawford R.W.,Kunz J.C. Application of AI to Triplequadrupole Mass Spectrometry // Proc. Nuclear Science Symposium, Institute of Electrical and Electronics Engineers.-San Francisco (Calif.),1983.
- Wong et al., Wong C.M.,Lanning S. AI in Chemical Analysis // Energy and Technology Review,Lawrence Livermore Nat. Laboratory.-Feb. 1984.
- Woods et al., Woods W-A. Kaplan R-M. Speech Understanding Systems: Final Report,BBN Report N 3438.-Bolt Berunek and Newman Inc.-Cambridge (Mas.). Oct. 1976.-P. 347.
- Woods et al., Woods W-A.,Kaplan R-M.,Nash-Webber B. The Lunar Sciences Language System: Final Report // Technical Rep. 2378, Bolt Berunek and Newman Inc.-Cambridge (Mas.),1978.-P. 286.
- Wright et al., Wright J-M.,Fox M-S. SRL/1.5 User Manual / Robotics Institute,Carnegie-Mellon University.-Pittsburgh,Dec. 1983.
- Xiang et al., Xiang Z.,Brihari S-N.,Shapiro S-C. Analogical and Propositional Representations of Structure in Neurological Diagnosis // Proc. CAI Applications,IEEE Computer Society.-Dec.1984.
- Yan et al., Yan C-K.,Venetsanopoulos A-N.,Fillery E-D. An Expert System for Landmarking of Cephalograms // Proc. 6th Int. Workshop on Expert Systems and Their Applications,Avignon.- 28-30 Apr.1986.-P. 337-356.
- Zarri,1983 Zarri G-P. An Outline of the Representation

- and Use of Temporal Data in the Reseda System // Information Technology: Research and Development.-1983.-Vol. 2.
- Zarri,1984a Zarri G.P. Expert Systems and Information Retrieval: an Experiment in the Domain of Biographical Data Management // Int. J. Man-Machine Studies.-1984.-Vol. 20.
- Zarri,1984b Zarri G.P. Inference Techniques for Intelligent Information Retrieval // Proc. Law and Technology Conf.-Houston (Tex.),1984.
- Zenkin,1983 Zenkin A.A. Some Extensions of Pall's Theorem // Comput. and Mathematics with Appl. -1983.-Vol. 9,N 4.-P. 609-615.
- Zue et al., 1986 Zue V.W.,Lamel L.F. An Expert Spectrogram Reader: a Knowledge-Based Approach to Speech recognition // Proc. ICASSP.-Tokyo,1986.-P. 1977-1200.

Оглавление

Предисловие	5
Предисловие редактора	7
Глава 1. Особенности реализации ЕЯ-систем (Э. В. Попов, А. Б. Преображенский)	9
1.1. Назначение и область применения ЕЯ-систем	9
1.2. Обобщенная схема ЕЯ-системы	12
1.3. Методы реализации ЕЯ-систем	17
1.4. Настройка ЕЯ-систем	28
Глава 2. Состояние развития современных ЕЯ-систем (А. Б. Преображенский)	32
2.1. Основные классы ЕЯ-систем	32
2.2. Интеллектуальные вопрос-ответные системы	40
2.3. Системы общения с базами данных	42
2.4. Диалоговые системы решения задач	51
2.5. Системы обработки связных текстов	59
Глава 3. Отечественные коммерческие и промышленные ЕЯ-системы	65
3.1. Коммерческая система АИСТ (А. Б. Преображенский)	65
3.2. Коммерческая система ЛИНГВИСТ (А. Ф. Киселев, Э. В. Попов)	75
3.3. Промышленная система ДИСПУТ (Л. И. Микулич)	87
Глава 4. Теоретические основы систем речевого общения (Б. М. Лобанов)	95
4.1. Основные положения	95
4.2. Фонетическая и просодическая структуры речи	96
4.3. Акустические характеристики фонем	99
4.4. Информационная структура речевого сигнала	103
Глава 5. Принципы построения систем речевого общения (Б. М. Лобанов)	106
5.1. Классификация речевых процессоров	106
5.2. Формантный анализатор речевых сигналов	109
5.3. ДП-анализатор речевых команд	114
5.4. Формантный синтезатор речевых сигналов	116
5.5. Универсальный фонемный синтезатор речевых сообщений	121
Глава 6. Состояние разработки и производства систем речевого общения (Б. М. Лобанов)	125
6.1. Обзор промышленных систем речевого общения	125
6.2. Отечественные системы речевого общения	132
6.3. Перспективные зарубежные системы речевого общения	136
6.4. Тенденции применения средств речевого общения	138
Глава 7. Особенности построения и использования систем переработки визуальной информации (В. В. Александров, В. А. Анисимов, Н. Д. Горский, С. Н. Мысько)	140
7.1. Назначение, классификация и области применения	140
	459

7.2. Автоматизированные системы обработки изображений	142
7.3. Системы анализа изображений	146
7.4. Системы машинной графики	150
Глава 8. Состояние развития современных систем обработки визуальной информации (В. В. Александров, В. А. Анисимов, Н. Д. Горский, С. Н. Мысько)	153
8.1. Автоматизированные системы обработки изображений	153
8.2. Системы анализа изображений	162
8.3. Системы и пакеты машинной графики	172
Глава 9. Отечественные системы обработки визуальной информации	179
9.1. Пакет программ для распознавания рукописной, символьной и графической информации ГРАФИТ (А. М. Байков, Е. О. Кузин, А. Л. Ламис)	179
9.2. Система технического зрения для анализа сложных трехмерных сцен (В. П. Андреев)	184
9.3. Когнитивная интерактивная графическая система ДСТЧ (А. А. Зенкин)	190
Глава 10. Теоретические аспекты машинного перевода (А. Д. Бакулов, Н. Н. Леонтьева)	201
10.1. Назначение машинного перевода	201
10.2. Периодизация и классификация систем машинного перевода	203
10.3. Лингвистическое обеспечение систем машинного перевода	206
10.4. Математическое и программное обеспечение машинного перевода	212
10.5. Оценка систем машинного перевода	213
10.6. Перспективы развития систем машинного перевода	214
Глава 11. Современное состояние машинного перевода (Н. Н. Леонтьева, З. М. Шаляпина)	216
11.1. Общие сведения о современных системах машинного перевода	216
11.2. Промышленные системы машинного перевода	217
11.3. Развивающиеся системы машинного перевода	235
11.4. Экспериментальные системы машинного перевода	241
Глава 12. Отечественные системы машинного перевода (А. Д. Бакулов, Н. Н. Леонтьева, З. М. Шаляпина)	248
12.1. Три подхода к решению задач машинного перевода	248
12.2. Комплекс систем англо-русского и немецко-русского автоматического перевода АНРАП	249
12.3. Системы семейства ЭТАП	253
12.4. Система французско-русского автоматического перевода ФРАП	255
12.5. О перспективах развития отечественных систем машинного перевода	260
Глава 13. Особенности разработки и использования экспертных систем (Э. В. Попов)	261
13.1. Назначение и принципы построения экспертных систем	261
13.2. Классификация экспертных систем	265
13.3. Классификация инструментальных средств	269
13.4. Методология разработки экспертных систем	271
13.5. Этапы разработки экспертных систем	274
13.6. Взаимодействие инженера по знаниям с экспертом	282
13.7. Трудности разработки экспертных систем	284
13.8. Проблемы и перспективы	288
Глава 14. Состояние разработки инструментальных средств и экспертных систем (Б. С. Кирсанов, Э. В. Попов)	290
14.1. Характеристики инструментальных средств и экспертных систем	290
14.2. Состояние разработки инструментальных средств	294

14.3. Состояние разработки экспертных систем	312
Глава 15. Отечественные инструментальные средства для персональных ЭВМ и мини-ЭВМ	342
15.1. Исследовательская система СПЭИС (О. В. Ковригин)	342
15.2. Коммерческая гибридная система ЭКСПЕРТ (Б. С. Кирсанов, Э. В. Попов)	349
15.3. Коммерческая система ФИАКР (С. Ю. Соловьев, Г. Н. Соловьева)	362
Глава 16. Отечественные оболочки экспертных систем для больших ЭВМ (Б. С. Кирсанов, Э. В. Попов)	369
16.1. Общее описание коммерческой гибридной системы ЭКСПЕРТИЗА	369
16.2. Возможности режима приобретения знаний	371
16.3. Возможности режима консультации	379
16.4. Функционирование системы	384
Глава 17. Отечественные экспертные системы на персональных ЭВМ	388
17.1. Система СПЭИС/МОДИС (О. В. Ковригин)	388
17.2. Система ОБЕЗБОЛИВАНИЕ/ЭКСПЕРТ (Б. С. Кирсанов, М. Д. Шапот)	394
Глава 18. Отечественные экспертные системы на больших ЭВМ	399
18.1. Система ПЛОТИНА/ЭКСПЕРТИЗА (К. И. Ивашенко, Б. С. Кирсанов)	399
18.2. Промышленная система ДАМП/ЭКСПЕРТИЗА (М. М. Виньков, Б. Е. Демин, Б. С. Игнатов, В. Н. Игнатьев)	404
18.3. Система СОНЭТ/ЭКСПЕРТИЗА (Б. С. Кирсанов, С. В. Маслов)	407
18.4. Система ЛЕДИ-Z (З. В. Рахманова)	411
Список литературы	419

Справочное издание
ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ: В 3 кн.
Кн. 1. СИСТЕМЫ ОБЩЕНИЯ И ЭКСПЕРТНЫЕ СИСТЕМЫ

Под редакцией **Э. В. Попова**

Справочник

Заведующая редакцией **Г. И. Козырева**
Редакторы **Т. М. Любимова, Т. М. Толмачева**
Переплет художника **Н. А. Пашуро**
Художественный редактор **Н. С. Шеин**
Технический редактор **Т. Г. Родина**
Корректор **О. Е. Иванецкая**

ИБ № 2229

Сдано в набор 25.09.89	Подписано в печать 15.01.90		
Т-06012	Формат 60×88 ¹ / ₁₆	Бумага офсетная № 2	Гарнитура литературная
Печать высокая	Усл. печ. л. 28,42	Усл. кр.-отг. 28,42	Уч. изд. л. 41,56
Тираж 25 000 экз.	Изд. № 22834	Зак. № 6859	Цена 2 р. 40 к.

Издательство «Радио и связь». 101000, Москва, Почтамт, а/я 693

Ордена Октябрьской Революции и ордена Трудового Красного Знамени МПО «Первая
Образцовая типография» Государственного комитета СССР по печати. 113054, Москва,
Валовая, 28.