МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ИЗВЕСТИЯ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ
ПРИБОРОСТРОЕНИЕ
ИЗДАНИЕ САНКТ-ПЕТЕРБУРГСКОГО НАЦИОНАЛЬНОГО ИССЛЕДОВАТЕЛЬСКОГО УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
Журнал издается с января 1958 г.

ТОМ 57 **НОЯБРЬ 2014** **№ 11**

ТЕМАТИЧЕСКИЙ ВЫПУСК

МЕТОДОЛОГИЯ И ТЕХНОЛОГИИ ПРОАКТИВНОГО УПРАВЛЕНИЯ СЛОЖНЫМИ ОБЪЕКТАМИ

Под редакцией доктора технических наук, профессора Б. В. Соколова

СОДЕРЖАНИЕ

CONTENTS

THEMATICAL ISSUE

METHODOLOGY AND TECHNOLOGIES OF PROACTIVE CONTROL OVER COMPLEX OBJECTS

By Edition of B. V. Sokolov, Doctor of Technical Science, Professor

Editor-in-Chief E. B. Yakovlev

ПРЕДИСЛОВИЕ

В настоящее время проблема сложности является одной из центральных проблем управления современными и перспективными организационно-техническими объектами. Данная проблема по своему содержанию имеет много аспектов, в том числе такие, как сложность описания объекта управления и соответствующей системы управления в целом, сложность моделирования и прогнозирования их поведения, а также сложность принятия решений в системе управления. При этом применительно к сложным организационно-техническим объектам выделяют еще один аспект управления, а именно, управление сложностью (complexity management problem). Анализ показывает, что в этом случае целесообразно переходить к новой технологии управления, базирующейся на концепции проактивного управления, которое, в общем случае, многофункционально и включает в себя применительно к сложным объектам как функции целеполагания, планирования, регулирования, так и функции учета и контроля, мониторинга и координации. Само же проактивное управление объектами, в отличие от традиционно используемого реактивного управления, ориентированного на оперативное реагирование и последующее недопущение возможных нештатных и аварийных ситуаций, предполагает предотвращение возникновения указанных ситуаций за счет создания в соответствующей системе управления принципиально новых прогнозирующих и упреждающих возможностей при формировании и реализации управляющих воздействий, базирующихся на методах и технологиях системного (комплексного) моделирования.

К настоящему времени наука создала богатый методологический и методический аппарат, который может быть положен в основу существующих и перспективных технологий проактивного управления сложными объектами и позволяет успешно преодолевать трудности, связанные с воздействием факторов сложности в современном мире. В данном тематическом выпуске журнала представлены результаты исследований проблем многофункционального проактивного управления сложными объектами, полученные, в большинстве своем, сотрудниками ФГУН Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН) — ведущего научного учреждения Отделения нанотехнологий и информационных технологий (ОНИТ) РАН в Северо-Западном регионе РФ.

Заместитель директора СПИИРАН по научной работе
Заслуженный деятель науки РФ,
доктор технических наук, профессор
Б. В. СОКОЛОВ

PREFACE

Today the problem of complexity is one of the central problems in control and management over modern and perspective organization-technical objects. The problem includes a lot of aspects, such as complexity of description of both object under control and corresponding control system as a whole, complexity of modeling and prediction of their behavior, as well as complexity of decision making in the control system. As applied to complex organization-technical objects, an additional aspect of control is recognized, namely, the complexity management problem. Analysis brings out the advisability of conversion, in this case, to a new technology of control, based on the concept of proactive control. As applied to complex objects, the multifunctional control description generally includes functions of target designation, planning and scheduling, execution, as well as accounting and supervision, monitoring and coordination. In comparison to traditionally employed reactive control, oriented to operative response and consequent exclusion of possible extraordinary and emergency situations, the proactive control over objects presupposes prevention of the above accidents through the creation, in the control system, of fundamentally new predictive and proactive capabilities for control actions formation and realization based on methods and technologies of system (complex) modeling and simulation.

By now the science has developed a reach methodological and methodical foundation which may be used as a background of existing and perspective technologies of proactive control over complex objects and allows overcoming the difficulties related to the effects of present-day complexity factors. This thematic issue of the Journal presents results of investigations in proactive control over complex objects carried out, for the major part, at St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), the leading scientific institution of Department of Nanotechnologies and Information Technologies of RAS in North-West region of the Russian Federation.

Doctor of Technical Science, Professor
B. V. SOKOLOV,
Honored Scientist of the Russian Federation,
Deputy Director for R&D,
St. Petersburg Institute for Informatics and Automation of RAS

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПРОАКТИВНОГО УПРАВЛЕНИЯ СЛОЖНЫМИ ОБЪЕКТАМИ

УДК 519.8

М. Ю. Охтилев, Н. Г. Мустафин, В. Е. Миллер, Б. В. Соколов

КОНЦЕПЦИЯ ПРОАКТИВНОГО УПРАВЛЕНИЯ СЛОЖНЫМИ ОБЪЕКТАМИ: ТЕОРЕТИЧЕСКИЕ И ТЕХНОЛОГИЧЕСКИЕ ОСНОВЫ

Рассмотрены теоретические и технологические основы разрабатываемой прикладной теории проактивного управления сложными объектами, которая к настоящему времени получила практическую реализацию в ракетно-космической отрасли, атомной энергетике, транспортно-логистической и военной сферах.

Ключевые слова: *междисциплинарный подход, управление сложностью, проактивный мониторинг и управление, комплексное моделирование.*

Введение. Анализ основных проблем XXI века показывает, что наиболее актуальной является проблема обеспечения безопасности жизнедеятельности в условиях возникновения различных масштабных аварий, техногенных катастроф и других чрезвычайных ситуаций, которые без оперативного принятия специальных мер могут привести к большим человеческим жертвам, материальным потерям и многим другим негативным факторам [1—3]. Одна из главных причин возникновения перечисленных явлений связана с усилением сложности существующих и проектируемых организационно-технических систем, используемых в различных предметных областях. При этом говоря о сложности современных объектов-оригиналов (реальных и абстрактных), принято выделять следующие основные аспекты: *структурную сложность, сложность функционирования, сложность принятия решений и выбора сценариев поведения, сложность развития, сложность их формального описания и моделирования* [1, 4, 5—11].

В этих условиях для обеспечения требуемой степени автономности, качества и оперативности управления сложными объектами (далее — объектами) необходимо, ***во-первых*****,** обеспечить модельно-алгоритмическое описание процессов смысловой интерпретации всех возможных штатных и нештатных состояний при их функционировании и, ***во-вторых*****,** на этой основе решить весь перечень задач комплексной автоматизации и интеллектуализации процессов управления объектами в различных условиях.

Однако, к сожалению, в подавляющем большинстве случаев на практике процессы мониторинга и управления состояниями объектов в указанной выше трактовке автоматизированы лишь частично [2, 6, 12]. Как правило, в современных автоматизированных системах управления (АСУ) сложными объектами операторам предоставляется смысловая информация только о состояниях их *элементов*, а не *объектов контроля в целом*. Указанные обстоятельства приводят к тому, что *интегральная оценка* состояния объектов в таких системах, как и формирование необходимых управляющих воздействий, осуществляется операторами

в основном вручную на базе тех или иных эвристических правил. Кроме того, как следует из вышеизложенного, особенность создания рассматриваемых объектов, систем и комплексов заключается в том, что они, прежде всего, должны быть ориентированы на применение в условиях возникновения неисправностей, аварий и даже катастроф и, следовательно, наделены свойством *живучести* (в более общем случае — катастрофоустойчивости) [3, 13, 14]. Применительно к процессам мониторинга и управления реализация свойства живучести предполагает оперативное формирование таких процедур сбора, обработки и анализа данных, а также соответствующей вычислительной среды, при которых обнаружение, локализация и ликвидация сбоев и отказов элементов и подсистем данных объектов будет происходить значительно раньше, чем проявятся возможные последствия указанных неисправностей. В этом и состоит основное содержание рассматриваемых в настоящей статье задач синтеза технологий проактивного (упреждающего) мониторинга и управления, которые можно рассматривать как перспективные технологии управления сложностью (complexity management) [1, 4—6, 8, 10, 11].

Следует отметить, что процессы проактивного управления объектами характеризуются дополнительными особенностями в условиях, когда из-за дефицита ресурсов (вызванного различными причинами субъективного и объективного характера) становится *невозможным поддерживать требуемый уровень работоспособности объектов.* В данных ситуациях проактивное управление соответствующими объектами и системами должно сопровождаться целенаправленными процедурами реконфигурации структур как самих объектов, так и АСУ объектами для обеспечения максимально допустимого уровня их работоспособности.

Анализ показывает [3, 5, 6, 12, 13, 15—19], что в качестве методологической и методической базы для решения перечисленных выше проблем целесообразно выбрать прикладную теорию проактивного мониторинга и управления структурной динамикой сложных объектов. Проактивное управление объектами, в отличие от традиционно используемого на практике реактивного управления (которое ориентировано на оперативное реагирование на инциденты и последующее их недопущение), предполагает предотвращение возникновения инцидентов за счет создания в соответствующей системе мониторинга и управления принципиально новых упреждающих возможностей при формировании управляющих воздействий на основе реализации концепции системного (комплексного) моделирования [6, 12, 16, 18, 20, 21].

К настоящему времени наука создала богатый методологический и методический аппарат, в основу которого положена междисциплинарная отрасль системных научных знаний. В ядре этой отрасли знаний, прежде всего, выделяют такие научные направления, как ***кибернетика*** *(в современных условиях —* ***неокибернетика), информатика и общая теория систем*** [1, 2, 6, 10, 21, 22]. Формирование отрасли системных научных знаний является велением времени, так как на данном этапе развития науки (этапе интеграции научных знаний) на передний план выступает методология, требующая сочетания (единства) процессов анализа и синтеза при изучении свойств объектов как целостных образований, состоящих из взаимосвязанных частей и обладающих качественно новыми свойствами по сравнению со свойствами этих частей. При этом в настоящее время речь должна идти не о взаимном поглощении, а о взаимном дополнении, концептуальном и идейном взаимообогащении, гармоничном и согласованном развитии междисциплинарных наук. В данной статье на примере разрабатываемой авторами теории проактивного мониторинга и управления структурной динамикой объектов иллюстрируются указанные тенденции.

Теоретические основы проактивного управления сложными объектами. Анализ современного состояния фундаментальных и прикладных научных работ в области решения проблем управления сложностью показал, что время реакции на происходящие в этой области перемены, вызванные научно-техническим прогрессом, и адаптации к ним теоретических исследований значительно превышает интервал между очередными изменениями [2, 4, 5, 9—11, 17]. Все это требует проведения упреждающих исследований, основанных на прогнозировании

возможных проблем в рассматриваемой предметной области и разработке соответствующих методологических и методических основ их решения.

При этом в ряде работ [1, 4, 5, 10, 15] подчеркивается глубокая общность биологических объектов, современных АСУ объектами и корпоративных информационных систем (ИС) вследствие их иерархически-сетевой организации. Разрабатываемые в настоящее время архитектуры, ориентированные на сервисы и базирующиеся на концепции виртуализации своих компонентов, обеспечивают материальную основу для синтеза принципиально новых информационно-вычислительных и телекоммуникационных систем, которые по своим свойствам будут приближаться к свойствам живых организмов.

Одним из классиков современной кибернетики С. Биром в работе [1] было показано, как на основе нейрофизиологической интерпретации функционирования центральной нервной системы человека удается построить оригинальную пятиуровневую модель жизнеспособной системы, в которой за счет гибкого сочетания механизмов иерархического и сетевого управления можно находить необходимый (в зависимости от складывающейся ситуации) компромисс между централизацией и децентрализацией целей, функций, задач и операций, реализуемых в соответствующей организации и определяющих ее специфику.

Данную модель С. Бир успешно использовал при решении различных классов задач прогнозирования и анализа путей развития сложных социально-экономических систем [1]. При этом в своих работах С. Бир неоднократно подчеркивал, что конструктивное исследование многоаспектной проблемы сложности должно базироваться на дальнейшем диалектическом развитии принципа необходимого разнообразия, сформулированного Р. Эшби. Анализ работ [2, 4—7, 13—16, 20, 21] в области современной кибернетики (неокибернетики) позволил сформулировать ряд конкретных направлений по реализации данного принципа (см. рис. 1), которые могут быть положены в основу предлагаемой авторами концепции проактивного управления сложными объектами.

Рис. 1

В работах [6, 10, 12, 13, 16, 18—21] перечисленные направления реализации принципа необходимого разнообразия получили свою дальнейшую конкретизацию и развитие для ряда предметных областей. Авторами данных работ подчеркивается особая актуальность создания методологических и методических основ решения проблем *управляемой самоорганизации* как наиболее эффективного способа борьбы с разнообразием состояний внешней среды. При

этом технология управляемой самоорганизации предполагает реализацию целенаправленных процессов поддержания динамического соответствия структур и функций в соответствующих сложных организационно-технических и социально-экономических системах. К настоящему времени получен ряд интересных теоретических и практических результатов при исследовании проблем управления структурной динамикой сложных технических объектов в различных предметных областях [5, 6, 9—12, 18].

В современных условиях существуют различные варианты организации проактивного управления сложными объектами, в том числе и технологии проактивного управления структурной динамикой указанных объектов. Среди данных технологий можно выделить в первую очередь изменение способов и целей функционирования объектов, последовательности выполнения операций, входящих в указанные технологии, в различных условиях; перемещение в пространстве отдельных элементов и подсистем объектов; перераспределение и децентрализацию функций, задач, алгоритмов управления и информационных потоков между уровнями структур объектов; использование гибких (сокращенных) технологий управления объектами; реконфигурацию структур объектов при их деградации [6].

Задачи управления структурной динамикой объектов по своему содержанию относятся к классу задач структурно-функционального синтеза облика объектов и формирования соответствующих программ управления их развитием. Главная трудность и особенность решения задач рассматриваемого класса состоит в следующем. Оптимальные программы управления основными элементами и подсистемами объекта могут быть выполнены лишь после того, как станет известен перечень функций и алгоритмов обработки информации и управления, которые должны быть реализованы в указанных элементах и подсистемах. В свою очередь, распределение функций и алгоритмов по элементам и подсистемам объекта зависит от структуры и параметров законов управления данными элементами и подсистемами. Трудность разрешения данной противоречивой ситуации усугубляется еще и тем, что под действием различных причин во времени изменяется состав и структура объекта на разных этапах его жизненного цикла.

К настоящему времени рассматриваемый класс задач структурно-функционального синтеза и управления развитием объектов исследован недостаточно глубоко. Получены новые научные и практические результаты в рамках следующих направлений исследований [2, 6, 15,]: синтез технической структуры объекта при известных законах функционирования его основных элементов и подсистем; синтез функциональной структуры объекта или, иными словами, синтез программ управления его основными элементами и подсистемами при известной технической структуре объекта; синтез программ создания и развития новых поколений объектов без учета этапа совместного функционирования существующих и внедряемых объектов. Известен ряд итерационных процедур получения совместного решения задач, исследования которых проводятся в рамках указанных направлений. В целом, все существующие модели и методы структурно-функционального синтеза облика объектов и формирования программ их развития используются на этапах внешнего и внутреннего проектирования облика, т.е. когда фактор времени не является существенным.

В рамках разработанного авторами подхода к организации проактивного управления объектами удалось с единых позиций подойти к решению всего спектра задач их структурно-функционального синтеза, возникающих на различных этапах жизненного цикла. Динамическая и управленческая интерпретация указанных задач, а также реализация концепции комплексного моделирования позволили на конструктивном уровне использовать фундаментальные и прикладные результаты, полученные к настоящему времени в таких научных дисциплинах, как исследование операций, искусственный интеллект, теория управления, теория принятия решений, системный анализ.

В заключение данного раздела приведем в качестве примера содержание предложенной авторами обобщенной процедуры решения задачи проактивного управления структурной ди-

намикой объекта, в соответствии с которой на первом этапе должно осуществляться формирование (генерирование) допустимых вариантов многоструктурных макросостояний объекта или, другими словами, должен проводиться структурно-функциональный синтез его нового облика, соответствующего складывающейся (прогнозируемой) обстановке.

На втором этапе производится выбор и реализация конкретного варианта многоструктурного макросостояния объекта с одновременным синтезом (построением) адаптивных планов (программ) управления его переходом из текущего в требуемое (выбранное) макросостояние. При этом рассматриваемые планы должны обеспечивать такое эволюционное развитие объекта, при котором наряду с реализацией программ перехода из соответствующих макросостояний предусматривается одновременно и реализация программ устойчивого управления объектом в промежуточных макросостояниях. В целом, на втором этапе исследования задачи выбора оптимальных программ проактивного управления структурной динамикой объекта приходится решать совокупность частных задач многоуровневой и многоэтапной оптимизации.

Одно из главных достоинств предлагаемой процедуры поиска и реализации оптимальных программ проактивного управления структурной динамикой объекта состоит в том, что при формировании вектора программных управлений в результате, наряду с оптимальным планом, одновременно получаем и искомое многоструктурное макросостояние, находясь в котором, объект сможет выполнять поставленные перед ним задачи в складывающейся (прогнозируемой) обстановке с требуемой степенью устойчивости.

В результате проведенных исследований были разработаны комбинированные методы и алгоритмы решения задачи выбора указанных оптимальных программ в централизованном и децентрализованном режимах функционирования объекта [6, 12, 16, 18, 21]. В качестве базового комбинированного метода предложено использовать сочетание метода ветвей и границ и метода последовательных приближений. Теоретическим обоснованием данного метода служит доказанная теорема о свойствах задачи выбора оптимальной программы проактивного управления структурной динамикой объекта в условиях снятия ряда ограничений.

Технологические основы проактивного управления сложными объектами. Анализ современных тенденций развития информационных технологий и систем (ИТ и ИС) показывает, что все ведущие зарубежные и отечественные компании, специализирующиеся в данной области, строили и строят корпоративные информационные инфраструктуры только по вертикальному принципу, руководствуясь частными критериями и плохо согласуя собственные представления с требованиями бизнеса [4, 8—11, 14]. В результате традиционные подходы к автоматизации бизнес-процессов находятся в настоящее время если не в кризисном, то в предкризисном состоянии. При этом трудности управления современными АСУ объектами, а также корпоративными ИС выходят за рамки администрирования отдельными программными средами. Необходимость интеграции нескольких гетерогенных сред в общекорпоративные вычислительные системы и стремление выйти за пределы компании, подключившись к сети Интернет, обусловливают формирование нового уровня сложности.

Для преодоления указанных тенденций весьма перспективным представляется создание новых поколений ИТ и ИС, построенных на основе концепций ***адаптивного управления*** и ***самоорганизации***. Разрабатываемые ***самоуправляемые вычислительные системы***, по замыслам их создателей, должны в будущем самостоятельно организовывать свое функционирование с учетом требований, сформулированных администраторами. Говоря о свойствах будущих адаптивных и самоорганизующихся компьютерных систем, необходимо, в первую очередь, выделить следующие свойства [4—7, 10, 11, 14]: самосознание и проактивность; способность к переконфигурированию (самоконфигурирование); самосовершенствование и самооптимизация; самолечение; самосохранение; общественное поведение; коммуникабельность; благожелательность и правдивость.

В современных условиях ведущие производители компьютерных технологий и систем осознают необходимость и важность проблем создания и внедрения концепции адаптивного проактивного управления и самоорганизации в информационную сферу. Информационные технологии XXI века уже получили определение „естественные“, „органичные“ (Organic IT). Данной терминологией аналитики компании “Forrester Research” (США) [4, 14] хотят подчеркнуть необходимость более органичного, естественного, непосредственного использования информационных технологий в интересах бизнес-приложений.

Среди крупных корпораций-производителей информационных услуг, осуществляющих продвижение к „естественным“ компьютерным системам, можно, в первую очередь, назвать следующие [4, 14]: Dell-Dynamic Computing, Hewlett-Packard-Adaptive Infrastructure (Adaptive Enterprise), IBM-Computing on Demand, Autonomous Computing, Microsoft-Dynamic Systems, Sun Microsystems-N1 (все — США).

Разработчики отечественной концепции проактивного управления объектами в качестве стратегической цели (миссии) определили формирование методологии ***обеспечения технологической независимости*** от зарубежных производителей в области создания, эксплуатации и модернизации модельно-алгоритмического, технического, информационного и программного обеспечения процессов комплексной автоматизации и интеллектуализации. Данная методология должна базироваться на ***принципиально новом подходе*** к проектированию и применению соответствующих АСУ объектами, основанном на комбинированном использовании логических, лингвистических и математических моделей, методов и алгоритмов, обеспечивающих суперкомпьютерную обработку и анализ в реальном времени сверхбольших объемов информации при наличии в ней некорректных, неточных и противоречивых данных [6].

При этом новизна разработанной теории проактивного управления объектами состоит в том, что ее авторам удалось, базируясь на сформулированных ими концепциях управляемой структурной динамики и инвариантности состояний объектов, а также состояний распределенного асинхронного вычислительного процесса, их описывающих, осуществить переход от *эвристических* методов алгоритмизации этих процессов *к последовательности целенаправленных теоретически и методически обоснованных и взаимосвязанных этапов* построения как *алгоритмов анализа многоструктурных макро- и микросостояний объектов, так и алгоритмов проактивного управления ими*.

На рис. 2 представлены основные принципы построения системы проактивного управления сложными объектами, которые к настоящему времени получили широкую и всестороннюю реализацию в ракетно-космической отрасли, атомной энергетике, траспортно-логистической и военной сферах.

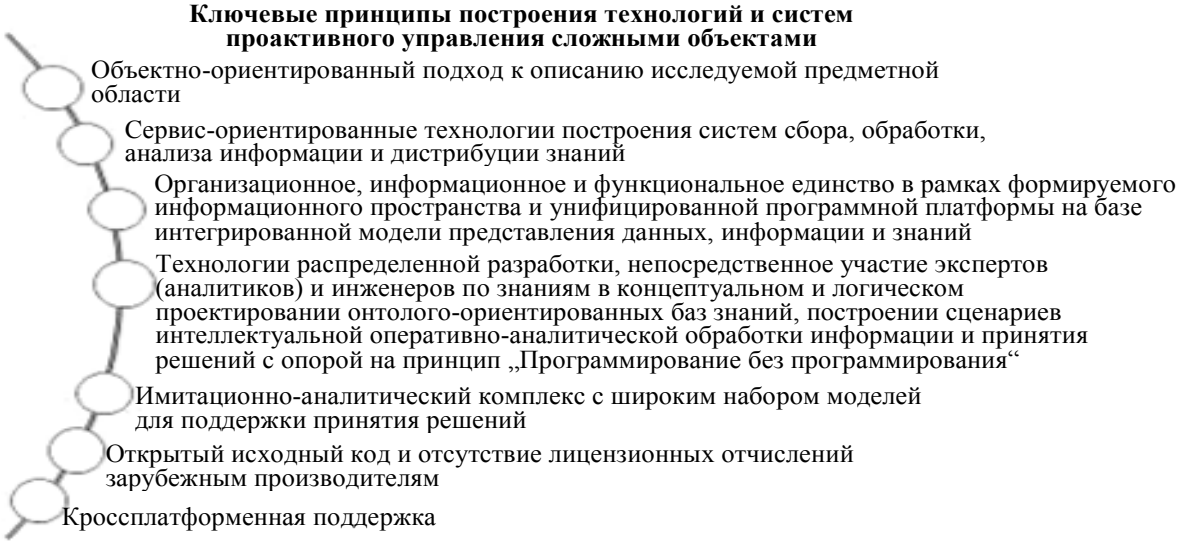


Рис. 2

Заключение. За прошедшие 20 лет теория проактивного мониторинга и управления структурной динамикой сложных объектов развивалась в рамках следующих трех основных научных направлений: разработка методологических и методических основ решения проблем адаптивного структурного-функционального синтеза и проактивного управления объектами; квалиметрия моделей и полимодельных комплексов, описывающих объекты на различных этапах их жизненного цикла; разработка и реализация инструментальных средств автоматизации и интеллектуализации процессов комплексного моделирования, прогнозирования, мониторинга состояний объектов в различных условиях.

Начиная с 1999 г. полученные фундаментальные и прикладные результаты повсеместно внедряются в организациях и учреждениях РАН, государственных и коммерческих организациях РФ, зарубежных организациях. Это позволило, в частности, решать вопросы прогнозирования при планировании и проектировании таких сверхсложных систем, как объекты воздушно-космической обороны [23]. Принятие решения о будущих угрозах и, как следствие, о возможных способах реакции на них путем разработки новых средств соответствующей информационной системы и (или) модернизации существующих средств является одним из характерных приложений теории проактивного управления сложными объектами. Прикладное значение рассматриваемой теории существенно возрастает в условиях нестабильной геополитической обстановки, когда сохраняется значимость стратегических решений, но особенно актуальными становятся тактические (в геополитическом смысле) решения, позволяющие оперативно реагировать на внешние воздействия.

К числу наиболее значимых результатов можно отнести следующие.

— Разработаны методологические и методические основы решения задач структурно-функционального синтеза интеллектуальных информационных технологий и систем управления объектами, базирующиеся на полимодельном многокритериальном описании, полученном в рамках теории недоопределенных вычислений и управления структурной динамикой. Предлагаемый подход позволил осуществлять в интерактивном либо автоматическом режиме интеллектуальную обработку данных и знаний о состоянии объектов, разнотипных по своей физической природе и формам представления, а также при наличии некорректной и недостоверной информации.

— Разработаны основы теории управления структурной динамикой объектов, содержащие концепции, принципы, способы, методы, алгоритмы и методики управления структурной динамикой. Данная прикладная теория имеет междисциплинарный характер и базируется на результатах, полученных в таких областях, как классическая теория управления, исследование операций, искусственный интеллект, теория систем и системный анализ.

— Разработаны основные понятия, принципы и подходы, используемые в квалиметрии семиотических моделей (полимодельных комплексов). Построена иерархия концептуальных моделей развивающихся ситуаций, участниками которой являются субъекты и объекты моделирования, а также собственно разрабатываемые (используемые) модели. Проведена классификация и систематизация семиотических моделей, установлены взаимосвязи и соответствия между различными их видами и родами. В рамках реализации концепции новых информационных технологий и разработанной методологии моделирования сложных объектов на основе алгоритмических сетей предложен методический подход, обеспечивающий разработку соответствующих систем автоматизации и моделирования и позволяющий пользователям оперативно и с минимальными трудозатратами строить и исследовать сетевые модели сложных объектов для различных предметных областей.

Исследования, выполненные по данной тематике, проводились при финансовой поддержке ведущих университетов Российской Федерации: Санкт-Петербургского государственного политехнического университета (мероприятие 6.1.1), Университета ИТМО (субсидия 074–U01), Программы научно-технического сотрудничества Союзного государства „Мониторинг СГ“

(проект 1.4.1–1), Российского фонда фундаментальных исследований (гранты № 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120, 13-06-0087), Программы фундаментальных исследований ОНИТ РАН (проект № 2.11), проектов ESTLATRUS 2.1/ELRI-184/2011/14, 1.2/ELRI-121/2011/13.

СПИСОК ЛИТЕРАТУРЫ

1. *Бир С.* Мозг фирмы. М.: Едиториал УРСС, 2005.
2. *Герасименко В. А.* Информатика и интеграция в технике, науке и познании // Зарубежная радиоэлектроника. 1993. № 5. С. 22—42.
3. *Панкратова Н. Д., Курилин Б. И.* Концептуальные основы системного анализа рисков в динамике управления безопасностью сложных систем // Проблемы управления и информатики. 2000. № 6. С. 110—132; 2001. № 2. С. 108—126.
4. *Вонт Р., Перинг Т., Тенненхау Д.* Адаптивные и проактивные компьютерные системы // Открытые системы. 2003. № 7. С. 4—9.
5. *Крылов С. М.* Неокибернетика: Алгоритмы, математика эволюции и технологии будущего. М.: Изд-во ЛКИ, 2008.
6. *Охтилев М. Ю., Соколов Б. В., Юсупов Р. М.* Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов. М.: Наука, 2006.
7. *Хиценко В. Е.* Самоорганизация: элементы теории и социальные приложения. М.: КомКнига, 2005.
8. *Foerster von H.* Cybernetics of Cybernetics: Paper Delivered at 1970 Annual Meeting of the American Society for Cybernetics / Univ. of Illions, Urbana, 1974.
9. *Foerster von H.* Cybernetics. Encyclopedia of Artificial Intelligence. N.Y.: John Wiley and Sons, 1987.
10. *Heikki Hyötyniemi.* Neocybernetics in Biological Systems / Helsinki Univ. of Technology, Control Engineering Laboratory, Rep. 151, Aug. 2006. 273 p.
11. *Maruyama M.* The second cybernetics. Deviation amplifying mutual causal process // American Scientist. 1963. N 51.
12. *Соколов Б. В., Юсупов Р. М.* Комплексное моделирование функционирования автоматизированной системы управления навигационными космическими аппаратами // Проблемы управления и информатики. 2002. № 5. С. 103—117.
13. *Колесников А. А.* Синергетические методы управления сложными системами: Теория системного синтеза. М.: КомКнига, 2006.
14. *Черняк Л.* От адаптивной инфраструктуры — к адаптивному предприятию // Открытые системы. 2004. Окт., № 9. С. 30—35.
15. *Васильев С. Н.* От классических задач регулирования к интеллектному управлению // Теория и системы управления. 2001. № 1. С. 5—22; № 2. С. 5—21.
16. *Калинин В. Н., Соколов Б. В.* Многомодельный подход к описанию процессов управления космическими средствами // Теория и системы управления. 1995. № 1. С. 56—61.
17. *Красовский А. А.* Науковедение и состояние современной теории управления техническими системами // Изв. РАН. Теория и системы управления. 1998. № 6. С. 16—24.
18. *Соколов Б. В., Юсупов Р. М.* Концептуальные основы оценивания и анализа качества моделей и полимодельных комплексов // Теория и системы управления. 2004. № 6. С. 5—16.
19. *Тимофеев А. В., Юсупов Р. М.* Интеллектуальные системы управления // Изв. РАН. Техническая кибернетика. 1994. № 5.
20. *Юсупов Р. М.* К 90-летию академика Е. П. Попова // Информационно-управляющие системы. 2005. № 1. С. 51—57.
21. *Юсупов Р. М., Соколов Б. В.* Проблемы развития кибернетики и информатики на современном этапе // Кибернетика и информатика: Сб. СПб: Изд-во СПбГПУ, 2006. С. 6—21.

22. *Винер Н.* Кибернетика или управление и связь в животном и машине. М.: Сов. радио, 1958.

23. *Охтилев М. Ю., Соколов Б. В., Юсупов Р. М.* Возможный подход к созданию единой информационно-вычислительной среды для системы воздушно-космической обороны // Вопросы оборонной техники: науч.-техн. сб. 2010. Сер. 9, вып. 1(242)—2(243). С. 85—90.

Сведения об авторах

Михаил Юрьевич Охтилев — д-р техн. наук, профессор; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; E-mail: oxt@mail.ru

Николай Габдрахманович Мустафин — канд. техн. наук, доцент; Санкт-Петербургский государственный электротехнический университет „ЛЭТИ“, кафедра автоматизированных систем обработки информации и управления; E-mail: nikolay.mustafin@gmail.com

Владимир Евгеньевич Миллер — канд. техн. наук; ОАО Радиотехнический институт им. акад. А. Л. Минца, Санкт-Петербург; директор филиала; E-mail: miller@progsystema.ru

Борис Владимирович Соколов — д-р техн. наук, профессор; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; зам. директора по научной работе; E-mail: sokol@iias.spb.su

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 519.872

Ю. И. Рыжиков

ОПТИМИЗАЦИЯ МАРШРУТНОЙ МАТРИЦЫ В СЕТЯХ ОБСЛУЖИВАНИЯ

Описан алгоритм расчета временных характеристик разомкнутой сети обслуживания. Предложен метод оптимизации сети обслуживания по среднему времени пребывания заявки в сети путем выравнивания загрузки узлов. Приводятся и обсуждаются результаты численного эксперимента.

Ключевые слова: *разомкнутая сеть, время пребывания, выравнивание загрузки узлов.*

Расчет сети. Реальные процессы обслуживания связаны с прохождением нескольких его этапов, реализуемых в отдельных узлах сети. Сеть обслуживания состоит из рабочих узлов, пронумерованных от 1 до M, источника (узел „0“) и стока (узел „M+1“). Для каждого j-го узла задаются моменты распределения „чистой“ длительности обслуживания $\{b_{j,l}\}$, $l = \overline{1, L}$, число каналов n_j и дисциплина обслуживания. Маршрут заявки в сети определяется неразложимой матрицей передач $R = \{r_{i,j}\}$, $i, j = \overline{0, M+1}$, образованной вероятностями перехода из i-го узла в j-й. Важнейшей оперативной характеристикой работы сети является среднее время пребывания в ней заявки. Первым шагом процесса оптимизации сети должна быть минимизация этого времени.

Проблема расчета сетей обслуживания активно обсуждается в сотнях статей и монографий (см., например, список литературы в работе [1]). К концу 1980-х гг. выяснилось, что строгое решение этой задачи возможно лишь при весьма ограниченных условиях теоремы BCMP (Baskett, Chandy, Muntz, Palacios [2]). Методы решения были непомерно трудоемкими, а получаемые характеристики — недостаточными. Как отмечал в ходе дискуссии на

конференции 1983 г. [3] П. Швейцер, „мы дошли до конца дороги с точными моделями...Мы затратили слишком много времени на такие модели, как мультипликативные сети и матрично-геометрические решения“. Альтернативой является только *потокоэквивалентная декомпозиция* сетей обслуживания.

В настоящей статье ограничимся рассмотрением разомкнутых однородных сетей с простейшими потоками. Последнее предположение базируется на известных теоремах о суммировании и случайном прореживании потоков. Интенсивности потоков определяются из уравнений баланса заявок:

$$\lambda_i = \Lambda r_{0,i} + \sum_{j=1}^{M} \lambda_j r_{j,i}, \quad i = \overline{1, M},$$

где Λ — суммарная интенсивность потока, поступающего из внешних источников.

Далее для всех узлов должно быть проверено условие отсутствия перегрузки $\lambda_i b_{i,1} / n_i < 1,$ обеспечивающее существование в сети стационарного режима.

С другой стороны, традиционное допущение о показательных распределениях времени обслуживания, как правило, является необоснованным, и порождаемые им ошибки могут быть сколь угодно велики. Это определяет целесообразность моделирования узлов сети системами с простейшим входящим потоком и произвольным распределением времени обслуживания. Последнее приходится аппроксимировать параллельно-последовательным набором фаз с экспоненциально распределенной задержкой в каждой. Приемлемую точность (выравнивание трех заданных моментов) обеспечивает, например, гиперэкспоненциальная аппроксимация с двумя составляющими. Заметим, что возможные случаи комплексных параметров и „парадоксальных“ вероятностей (одна отрицательна, а вторая больше единицы) не влияют на осмысленность конечных результатов. После такой аппроксимации расчет распределения числа заявок в узле можно выполнить итерационным методом Такахаси — Таками или методом матрично-геометрической прогрессии [4].

Для *разомкнутой* сети в целом среднее время пребывания заявки можно, как и для отдельного узла, вычислить на основе формулы Литтла

$$v = \sum_{i=1}^{M} \overline{k_i} / \Lambda$$

(среднее число заявок в сети k_i делится на суммарную интенсивность входящего потока), которая проверена многократно (в том числе, автором данной статьи).

Оптимизация сети. Эта задача в первом приближении решается как минимизация среднего времени пребывания заявки в сети и имеет множество аспектов: в частности, выбор производительности и количества обслуживающих устройств в узлах (см., например, [5]). Реально производительность таких устройств должна выбираться из *конечного ряда* значений, а количество устройств должно быть *целочисленным*. Поэтому постановка задачи о комплексной оптимизации сети *формальными* методами представляется непродуктивной. Иначе обстоит дело с маршрутной матрицей, оптимизация которой вообще не затрагивает аппаратную часть сети обслуживания и материализуется „бесплатно“. Именно с такой оптимизации и следует начинать. Кроме того, необходимо учитывать, что на маршрутную матрицу могут быть наложены ограничения, диктуемые технологическими и/или организационными соображениями.

Формула Литтла позволяет считать, что минимизация ожидаемого количества заявок в сети одновременно минимизирует среднее время пребывания в ней заявки. Естественно проводить оптимизацию маршрутной матрицы передач путем последовательной „расшивки“ узких мест сети, что достигается выравниванием ожидаемого числа заявок в узлах или

коэффициентов загрузки последних. Опишем алгоритм выравнивания (ради экономии места без разбивки на абзацы).

Рассчитать потоки на входе узлов; коэффициенты загрузки узлов; моменты распределения времени пребывания в узлах для однократного захода заявки; среднее количество q_i заявок в каждом узле сети и среднее время пребывания заявки в сети $T = \sum_{i=1}^{M} q_i / \Lambda$. Выбрать в качестве объекта разгрузки узел j^- с максимальным значением $q = q_{j-}$. Выбрать непосредственного „предшественника“ этого узла j^* минимум с двумя преемниками. Среди его преемников выбрать узел j^+ с наименьшим $q = q_{j+}$. Долю x потока интенсивностью $\lambda_{j^*} r_{j^*,j^-}$ переадресовать в узел j^+ посредством коррекции маршрутной матрицы.

Указанные действия выполняются, пока остаются значимыми уменьшения среднего времени пребывания заявки в сети. Коррекция всегда производится для ненулевых элементов матрицы передач, что позволяет запрещать недопустимые ребра маршрутов. Максимально допустимое значение x определяется из условия

$$\rho'_{j^+} + x\lambda_{j^*} r_{j^*,j^-} b_{j^+} / n_{j^+} <= 0,95,$$

где ρ'_{j^+} — исходный коэффициент загрузки.

Ниже рассматриваются два субоптимальных алгоритма оптимизации маршрутной матрицы.

Выравнивание числа заявок предполагает, что оптимальное значение x выбирается как абсцисса минимума параболы, аппроксимирующей зависимость от x суммарного числа заявок в узлах j^- и j^+. Парабола $Ax^2 + Bx + C$ строится по точкам для $x_0 = 0$, $x_1 = x_{\max}/2$ и $x_2 = x_{\max}$, причем слагаемые $x_0 = 0$ определяются по результатам первого этапа текущей итерации. Координата минимума параболы

$$x^* = \frac{x_1^2(y_2 - y_0) - x_2^2(y_1 - y_0)}{2[x_1(y_2 - y_0) - x_2(y_1 - y_0)]}.$$

В примере сети с шестью рабочими узлами при исходной маршрутной матрице стартовое среднее время пребывания заявки в сети составляет 16,247, после первой итерации — 7,869, после второй — 6,821. Далее происходят осцилляции в диапазоне [6,80, 7,02].

Выравнивание коэффициентов загрузки узлов. Приравнивая правые части выражений для новых коэффициентов загрузки, приходим к условию

$$x = \frac{\rho_{j^-} - \rho_{j^+}}{\lambda_{j^*} r_{j^*,j^-} (b_{j^-} / n_{j^-} + b_{j^+} / n_{j^+})}.$$

Реализация этого подхода обеспечивает монотонное уменьшение целевого показателя, которое по шагам составляет 7,84; 1,27; 0,183; $3{,}86 \cdot 10^{-2}$; $1{,}02 \cdot 10^{-2}$; $2{,}94 \cdot 10^{-3}$; $2{,}58 \cdot 10^{-4}$; $7{,}71 \cdot 10^{-5}$; $6{,}87 \cdot 10^{-6}$. Последний результат составил 6,896.

Обсуждение результатов. Из сопоставления результатов следует:

— оба рассмотренных метода работоспособны, несложны, быстро (в примере — за три шага) приводят к практически приемлемому результату и обеспечивают значительное уменьшение среднего времени пребывания заявки в сети;

— минимизация суммарного числа заявок в сети после некоторого числа монотонных улучшений порождает осциллирующий процесс;

— выравнивание коэффициентов загрузки монотонно приводит практически к тому же результату (разница в третьем знаке) и в каждой итерации исключает необходимое для параболической аппроксимации дополнительное двукратное обращение к процедуре расчета модели $M / H_2 / n$ для двух узлов сети.

Таким образом, для реальных расчетов предпочтительно применять выравнивание коэффициентов загрузки узлов.

Предложенный подход можно обобщить и на неоднородный поток заявок.

Заключение. Алгоритм коррекции маршрутных матриц прост, эффективен и позволяет легко учесть ограничения на допустимость коррекции их отдельных элементов. Если работа с маршрутной матрицей не дает приемлемых результатов, следует использовать эту технологию в комбинации с последовательным повышением производительности наиболее загруженных узлов — увеличением числа каналов или их быстродействия. Такую оптимизацию следует вести в диалоговом режиме.

СПИСОК ЛИТЕРАТУРЫ

1. *Ивницкий В. А.* Теория сетей массового обслуживания. М.: Физматлит, 2004.

2. *Baskett F., Chandy K. M., Muntz R. R., Palacios J. G.* Open, closed, and mixed networks of queuing with different classes of customers //J. of the ACM. 1975. Vol. 22, N 2. P. 248—260.

3. Mathematic computer performance and reliability // Proc. of the Intern. Workshop, Piza, 1983. Amsterdam: North-Holland Publ. Co, 1984. 429 p.

4. *Рыжиков Ю. И.* Алгоритмический подход к задачам массового обслуживания: Монография. СПб: ВКА им. А. Ф. Можайского, 2013. 496 с.

5. *Янбых Г. Ф., Столяров Б. А.* Оптимизация информационно-вычислительных сетей. М.: Радио и связь, 1987. 232 с.

Сведения об авторе

Юрий Иванович Рыжиков — д-р техн. наук, профессор; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; Военно-космическая академия им. А. Ф. Можайского, кафедра математического обеспечения ЭВМ, Санкт-Петербург; профессор; E-mail: ryzhbox@yandex.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 004.896

В. Ю. Будков, А. Л. Ронжин

ИНФОРМАЦИОННАЯ МОДЕЛЬ СОПРОВОЖДЕНИЯ РАСПРЕДЕЛЕННЫХ МЕРОПРИЯТИЙ В ИНТЕЛЛЕКТУАЛЬНОМ ЗАЛЕ СОВЕЩАНИЙ

Рассматривается проблема информационно-технического сопровождения распределенных совещаний и проанализированы основные этапы организации мероприятий с территориально распределенными участниками. Предложена информационная модель, описывающая способы обработки и обмена потоками данных между удаленными участниками в зависимости от ситуации в интеллектуальном зале совещаний.

Ключевые слова: *интеллектуальное пространство, распределенные мероприятия, аудиовизуальная обработка данных, протоколирование дикторов, информационная значимость.*

Введение. Для распределенных мероприятий характерна ситуация, при которой часть участников находится в зале совещаний и имеются удаленные участники, использующие персональные и мобильные устройства для подключения к веб-системе трансляции совещаний. Системы сопровождения таких мероприятий получили наибольшее развитие с возникновением научной парадигмы окружающего интеллектуального пространства [1], обеспечивающего проактивное ненавязчивое персонифицированное обслуживание участников. Так как под окружающим интеллектуальным пространством понимается глобальное единое пространство, то его создание в ближайшее время затруднительно, поэтому сейчас ведутся исследования по разработке отдельных менее масштабных прототипов интеллектуальных пространств, например „умная“ комната, „умный“ дом, „умный“ город [2, 3].

Существующие исследовательские прототипы интеллектуальных залов представляют собой распределенную сеть аппаратно-программных модулей, активационных устройств, мультимедийных средств и аудиовизуальных сенсоров. С увеличением количества решаемых задач и обслуживаемых пользователей становится сложно контролировать множество программных и аппаратных модулей, задействованных в интеллектуальном пространстве, поэтому необходимо математическое обеспечение и программные средства, реализующие управление совместной работой распределенных модулей.

Анализ существующих систем сопровождения. При разработке систем сопровождения распределенных мероприятий выделяют несколько этапов, требующих автоматизации [4]: 1) организация совещания, где определяются основные участники и утверждается план совещания; 2) подготовка совещания, в ходе которой производится оповещение участников и проверка готовности их участия; 3) проведение совещания, включающее обсуждение и подготовку протокола; 4) завершение совещания, где утвержденный протокол рассылается участникам; 5) контроль решений совещания, включающий рассылку напоминаний и оценку выполнения решений; 6) анализ материалов совещания, собранных в ходе предыдущих этапов, с использованием базы данных мероприятия для поиска и просмотра необходимой информации.

Для выявления основных проблем существующих систем сопровождения распределенных мероприятий был проведен их сравнительный анализ по пяти типам характеристик: 1) входные модальности, используемые для анализа и записи поведения участников во время проведения совещания; 2) основные типы выходных данных, используемых при взаимодействии с пользователем системы; 3) основные виды применяемого оборудования; 4) сервисы

обработки аудиовизуальных данных, записанных в ходе мероприятия; 5) дополнительные возможности систем сопровождения.

Проанализируем кратко ряд систем сопровождения по приведенным выше характеристикам.

Система Webinar.ru служит для проведения онлайн-мероприятий с участием одного или нескольких выступающих и подключением до нескольких тысяч слушателей. Выступающие имеют возможность представлять слайды презентации, различные текстовые документы, а также свой рабочий стол. Система предоставляет возможность общения с использованием текстового чата. Система ориентирована на поддержку и проведение вебинаров, тренингов.

Система Cisco WebEx имеет, по сравнению с предыдущим сервисом, расширенные возможности в области обслуживания удаленных участников распределенных мероприятий: в частности, реализована поддержка мобильных устройств с операционными системами Android и IOS, поддержка VoIP-телефонии, а также применяются сенсорные панели для создания зарисовок от руки, которые транслируются другим участникам.

Система Openmeetings с открытым исходным кодом также служит для проведения онлайн-мероприятий. Передача данных производится с помощью сервера Red5. Система Openmeetings позволяет проводить онлайн-мероприятия с участием одного или нескольких выступающих без необходимости установки на компьютерах пользователей дополнительного программного обеспечения. В качестве клиентского приложения можно использовать обычный браузер. Система поддерживает совместную работу с офисными документами и совместное ведение записей, а также позволяет просматривать записи в различных форматах.

Система WebHuddle с открытым исходным кодом использует браузер для запуска клиентского приложения, написанного на языке Java. Система не поддерживает видеосвязь, но участники могут показывать презентационные материалы и передавать текстовые сообщения.

Meetecho — веб-ориентированная система, предназначенная для проведения распределенных мероприятий с использованием гетерогенных устройств. Система включает в себя набор инструментов, позволяющих делать зарисовки, передавать изображение с экрана пользователя, показывать презентационные материалы, проводить опросы среди участников мероприятия. Доступ к мероприятию может осуществляться с помощью планшетов и смартфонов под управлением систем Windows Mobile, Android и IOS.

Перечисленные системы Webinar.ru, Cisco WebEx, Openmeetings, WebHuddle, Meetecho обладают широкими функциональными возможностями в области телекоммуникаций, но проблемам автоматической обработки речи, анализа поведения участников во время диалога и другим вопросам, связанным с требованиями, предъявляемыми к информационным системам сопровождения распределенных мероприятий, уделено недостаточно внимания.

Основной целью разработки систем протоколирования формальных мероприятий является автоматизация процесса стенографирования выступлений участников. Однако автоматическое распознавание разговорной речи остается на сегодняшний день одной из основных нерешенных проблем в области речевых технологий. Процесс автоматического распознавания речи представляет собой преобразование акустического речевого сигнала в последовательность слов, которая затем может использоваться для анализа и интерпретации смысла речевого высказывания. Одним из возможных способов повышения точности распознавания речи является настройка дикторозависимых параметров системы автоматической обработки речи. Поэтому не менее важной задачей при обработке аудиозаписей выступлений является этап диаризации дикторов, обеспечивающий сегментацию реплик каждого диктора в одноканальном аудиосигнале и последующую группировку всех речевых фрагментов, относящихся к определенному диктору.

В задаче диаризации дикторов, в отличие от задачи аутентификации, число дикторов, участвующих в дискуссии, заранее неизвестно, и поэтому соответствующие модели речи дик-

торов необходимо создавать и обучать в процессе анализа записей автоматически, что существенно усложняет обработку речевого сигнала. Другим фактором, снижающим качество работы систем диаризации, является наличие в звуковом сигнале таких явлений, как „перекрывающаяся“ речь (когда одновременно разговаривают несколько людей), артефакты речи (чмокание, цокание языком) и невербальные паузы (кашель, смех), а также короткие реплики. Эксперименты показывают, что длительность „перекрывающейся“ речи в условиях конференции, совещания или деловой встречи может достигать 70 % от общего объема аудиозаписи. Кроме того, на качество записи существенно влияют особенности помещения, расположение дикторов и характеристики записывающей аппаратуры.

Рассмотрим информационную модель сопровождения распределенных мероприятий и ряд методов обработки мультимедийных сигналов, применяемых при трансляции совещаний.

Информационная модель сопровождения распределенных мероприятий. В предлагаемой модели сопровождения мероприятий выделяются три основных этапа (рис. 1). Разработанные для модели авторами настоящей статьи методы и программное обеспечение ориентированы на обработку мультимедийных сигналов, записываемых во время мероприятия, и подготовку отчетных материалов по его окончании [5—7].

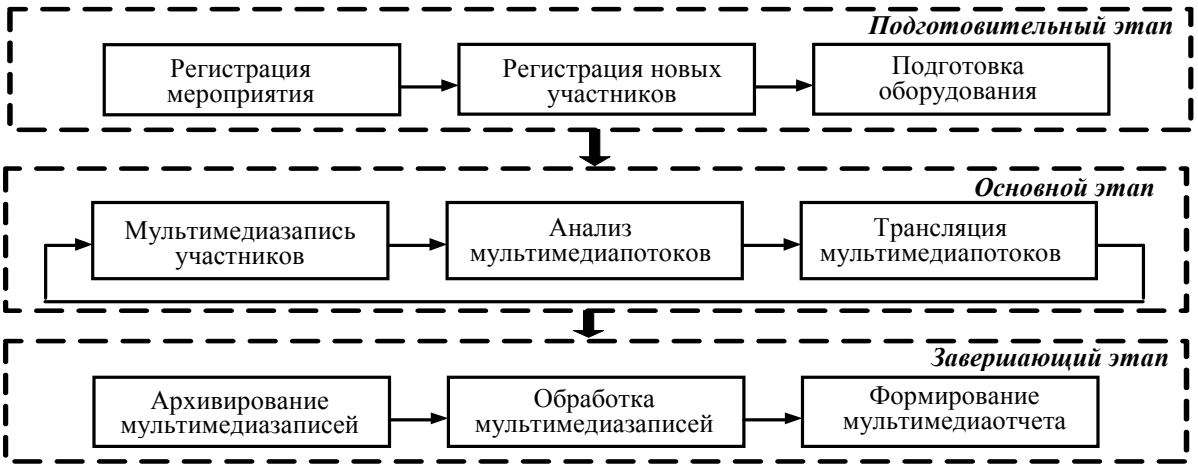


Рис. 1

Подготовительный этап включает в себя регистрацию мероприятия, подготовку оборудования и регистрацию участников. Вначале в систему сопровождения мероприятия вносятся следующие данные: время, место проведения, разрешения на доступ к трансляции мероприятия и т.д. После этого участники могут зарегистрироваться в системе и подписаться на участие в данном мероприятии. Перед началом мероприятия производится подготовка и настройка необходимого мультимедийного оборудования.

Основной этап включает в себя трансляцию и запись аудио- и видеопотоков и других мультимедийных данных, получаемых из источников, расположенных в зале совещаний, а также из удаленных участников. Выбор наиболее актуальной информации, которая транслируется удаленным участникам, производится на основе анализа текущей ситуации в зале.

На завершающем этапе сопровождения по окончании мероприятия производится анализ мультимедийных записей, их архивирование и создание отчета по мероприятию. Обработка данных, например идентификация участников, добавление новых не зарегистрированных ранее участников, ведется как в ручном, так и в автоматическом режиме. Формирование отчета производится по шаблонам, которые могут редактироваться вручную для получения необходимого формата.

Для описания предложенной модели сопровождения мероприятий введем следующие обозначения. На подготовительном этапе формируются основные сведения по предстоящему мероприятию: $M = <M_L, M_P, M_{Tb}, M_{Te}, M_U>$, где M_L — логотип мероприятия; M_P — список презентаций; M_{Tb}, M_{Te} — время начала и окончания мероприятия соответственно; M_U — множество

участников, включающее M_{U_inner} — множество участников, которые будут находиться в зале совещаний, и M_{U_outer} — множество удаленных участников, которые будут подключаться к дискуссии через сеть Интернет.

В ходе мероприятия производится формирование множества информационных потоков I, поступающих из источников нескольких типов, составляющих множество

$$ST=\{S_{video\_inner}, S_{video\_outer}, S_{audio\_inner}, S_{audio\_outer}, S_{projector}, S_{touch\_board}, S_{event\_server}\},$$

где S_{video_inner}, S_{audio_inner} — видеокамеры и микрофоны, установленные в зале; S_{video_outer}, S_{audio_outer} — видеокамеры и микрофоны, встроенные в клиентские устройства удаленных участников; $S_{projector}$ — проектор, установленный в зале; S_{touch_board} — сенсорная панель для рукописных записей, установленная в зале; S_{event_server} — центральный сервер, выдающий информацию о мероприятии, собранную на подготовительном этапе, и формирующий управляющие команды в ходе мероприятия; в зависимости от числа подключенных удаленных слушателей и оснащения зала имеется N источников данных: $\{S_1, S_2,\ldots, S_i,\ldots, S_N\} \in ST$.

Каждый источник S_i формирует информационный поток пакетов данных (I_i^1, I_i^2,… …, I_i^j,…, I_i^K), где K — число пакетов, полученных за время мероприятия. Каждый пакет содержит следующий набор параметров: $I_i^j=<D, f, t_b, t_e, w, S_i, u>$, где D — последовательность бинарных данных, f — формат передаваемых данных, t_b, t_e — время начала и окончания записи данных соответственно, w — частота дискретизации данных в пакете, S_i — источник данных, u — некоторый участник из множества M_U, предоставляющий текущий пакет данных. В предложенной модели сопровождения мероприятий использовались следующие форматы данных: $f \in$ {PCM, AVI, M-JPEG, VP8, H.264, BMP, JPEG, PNG, PPT, DOCX, TXT, BIN, CFG}, где первые десять являются стандартными для аудио- и видеоданных, изображений, презентаций и текстовых документов, а последние три служат для внутренней передачи служебных данных и конфигурационных параметров в текстовом и бинарном виде.

Выбор информационных потоков, используемых для передачи удаленным участникам, производится на основе событийной модели ситуации в зале. Множество событий включает следующие типы:

$$E = \{E_{participant\_act}, E_{participant\_talk}, E_{participant\_out}, E_{remote\_participant\_act}, E_{remote\_participant\_talk}, E_{remote\_participant\_out}, E_{projector\_act}, E_{new\_slide}, E_{slide\_obsolete}, E_{projector\_off}, E_{touchboard\_act}, E_{new\_sketch}, E_{sketch\_obsolete}, E_{touchboard\_off}, E_{participant\_sil}, E_{remote\_participant\_sil}\},$$

где $E_{participant\_act}$ — появление участника в зале, $E_{participant\_talk}$ — появление выступающего участника в зале, $E_{participant\_out}$ — выход участника из зала, $E_{remote\_participant\_act}$ — появление удаленного участника, $E_{remote\_participant\_talk}$ — появление выступающего удаленного участника, $E_{remote\_participant\_out}$ — отключение удаленного участника, $E_{projector\_act}$ — загрузка презентации, E_{new_slide} — появление нового слайда презентации, $E_{slide\_obsolete}$ — истечение максимального времени показа нового слайда презентации, $E_{projector\_off}$ — отсутствие презентации, $E_{touchboard\_act}$ — включение сенсорной доски, E_{new_sketch} — появление новой записи на сенсорной доске, $E_{sketch\_obsolete}$ — истечение максимального времени показа новой записи на сенсорной доске, $E_{touchboard\_off}$ — выключение сенсорной доски, $E_{participant\_sil}$ — отсутствие аудиоактивности участника, $E_{remote\_participant\_sil}$ — отсутствие аудиоактивности удаленного участника.

Каждое событие содержит данные об источнике и времени наступления события.

Учитывая, что при выводе мультимедийного контента на устройство удаленного участника могут быть использованы встроенные средства вывода только аудио- и видеоданных, в множестве ST можно выделить два подмножества, отвечающие за эти типы данных:

$$ST_{audio} = <S_{audio\_inner}, S_{audio\_outer}> \text{ и } ST_{video} = <S_{video\_inner}, S_{video\_outer}, S_{projector}, S_{touch\_board}, S_{event\_server}>.$$

В результате анализа поступивших событий из множества E в текущий момент времени может быть сформирован только один информационный поток, поступающий из источников

подмножества ST_{audio}, и выбрано несколько информационных потоков, поступающих из источников подмножества ST_{video}. Число одновременно отображаемых графических информационных потоков зависит от возможностей клиентского устройства.

Схема, отображающая метод формирования текущего мультимедийного контента на основе поступающих информационных потоков и анализа событий в зале совещаний, приведена на рис. 2.

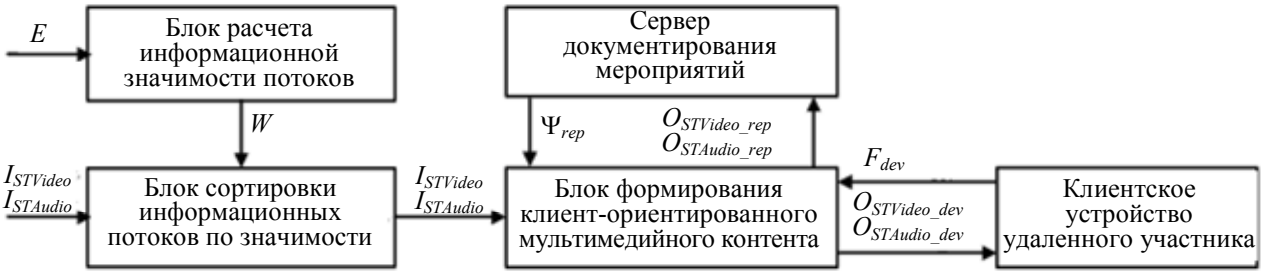


Рис. 2

На основе событий E производится расчет информационной значимости W мультимедийных потоков и сортировка пакетов I, формируемых устройствами аудиозаписи, проектором и сенсорной доской. Информационная значимость W каждого пакета I выбирается в зависимости от зарегистрированных событий, причем наличие аудиоактивности влияет на значимость аудио- и видеопотоков:

$$W_{video\_inner} = \begin{cases} 2, E_{participant\_act} \wedge E_{participant\_talk}, \\ 1, E_{participant\_act}, \\ 0, E_{participant\_out}; \end{cases} \quad W_{audio\_inner} = \begin{cases} 1, E_{participant\_talk}, \\ 0, E_{participant\_sil}; \end{cases}$$

$$W_{video\_outer} = \begin{cases} 2, E_{participant\_act} \wedge E_{participant\_talk}, \\ 1, E_{remote\_participant\_act}, \\ 0, E_{remote\_participant\_out}; \end{cases} \quad W_{audio\_outer} = \begin{cases} 1, E_{remote\_participant\_talk}, \\ 0, E_{remote\_participant\_sil}; \end{cases}$$

$$W_{projector} = \begin{cases} 3, E_{new\_slide}, \\ 2, E_{slide\_obsolete}, \\ 1, E_{projector\_act}, \\ 0, E_{projector\_off}; \end{cases} \quad W_{touch\_board} = \begin{cases} 3, E_{new\_sketch}, \\ 2, E_{sketch\_obsolete}, \\ 1, E_{touchboard\_act}, \\ 0, E_{touchboard\_off}. \end{cases}$$

На вход блока формирования мультимедийного клиентского контента поступают пакеты $I_{STVideo}$, $I_{STAudio}$, представленные в виде списка, упорядоченного по убыванию значения W, где с учетом характеристик F_{dev} клиентского устройства осуществляется компоновка контента из множеств выбранных видео- и аудиопакетов соответственно: $O_{STVideo\_dev}$, $O_{STAudio\_dev}$. Устройство характеризуется следующими параметрами: $F_{dev} = <F_{dev\_os}, F_{dev\_browser}, F_{dev\_resolution}, F_{dev\_connection\_speed}>$, где F_{dev_os} — операционная система устройства, $F_{dev\_browser}$ — используемый браузер, $F_{dev\_resolution}$ — разрешение экрана; характеристики F_{dev_os} и $F_{dev\_browser}$ отвечают за выбор формата передачи видеопотоков (VP8, M-JPEG, JPEG); за выбор качества видеопотока и частоты обновления кадров отвечает характеристика $F_{dev\_connection\_speed}$; на основе характеристики $F_{dev\_resolution}$ выбирается расположение и количество форм для вывода мультимедийного контента.

Например, в случае использования мобильного устройства с ограниченным размером экрана имеется возможность применить только одну форму для вывода видеоконтента, находящегося на первом месте в списке $I_{STVideo}$. Если разрешение экрана допускает размещение нескольких форм, то выводится подмножество видеопотоков, находящихся вверху этого

списка. Было апробировано четыре варианта расположения форм, при этом компоновка для максимального разрешения экрана содержала формы для вывода слайда презентации, рукописного наброска и видеоданных, записываемых видеокамерами, направленными на аудиторию, текущего выступающего, удаленных участников.

При формировании отчетной документации по мероприятию в зависимости от заданного сервером формата $\Psi_{rep} = <\Psi_{rep\_data}, \Psi_{rep\_data\_position}>$ производится генерация контента множеств $O_{STVideo\_rep}, O_{STAudio\_rep}$, где Ψ_{rep_data} содержит список необходимых типов пакетов данных, отображаемых в отчете, а $\Psi_{rep\_data\_position}$ — их расположение. При формировании отчета по мероприятию в оффлайн-режиме накладываются менее жесткие требования по скорости обработки данных, и могут быть привлечены средства автоматизированной обработки речи и текста [8—12].

Заключение. Предложенная информационная модель сопровождения участников распределенных мероприятий характеризуется применением средств автоматической обработки мультимедийных сигналов в целях автоматизации процесса трансляции и подготовки отчетных материалов по результатам мероприятия. В основу модели положен метод формирования текущего мультимедийного контента, использующий событийную модель анализа информационной значимости аудиовизуальных потоков при подготовке данных для трансляции удаленному участнику и отчетных материалов по мероприятию. Проведен сравнительный анализ функциональных характеристик существующих систем сопровождения веб-конференций и разработанной модели.

Статья подготовлена по результатам работы, выполняемой при поддержке Российского фонда фундаментальных исследований (проект № 13-08-0741-а).

СПИСОК ЛИТЕРАТУРЫ

1. *Zelkha E., Epstein B.* From Devices to Ambient Intelligence / Digital Living Room Conference. 1998. June.

2. *Юсупов Р. М., Ронжин А. Л.* От умных приборов к интеллектуальному пространству // Вестник РАН. 2010. Т. 80, вып. 1. С. 45—51.

3. *Aldrich F.* Smart Homes: Past, Present and Future / Inside the Smart Home / Ed. *R. Harper*. London: Springer-Verlag, 2003. P. 17—39.

4. Эффективные совещания [Электронный ресурс]: <http://am-meetingpoint.com/2013/02/16/effektivnye-soveshhaniya-podgotovka-provedenie-kontrol/>.

5. *Ронжин А. Л., Будков В. Ю.* Технологии поддержки гибридных е-совещаний на основе методов аудиовизуальной обработки // Вестник компьютерных и информационных технологий. 2011. № 4. С. 31—35.

6. *Ронжин А. Л., Карпов А. А., Кагиров И. А.* Особенности дистанционной записи и обработки речи в автоматах самообслуживания // Информационно-управляющие системы. 2009. Т. 5, вып. 42. С. 32—38.

7. *Ронжин Ал. Л., Будков В. Ю., Ронжин Ан. Л.* Формирование профиля пользователя на основе аудиовизуального анализа ситуации в интеллектуальном зале совещаний // Тр. СПИИРАН. 2012. Вып. 23. С. 482—494.

8. *Мещеряков Р. В.* Структура систем синтеза и распознавания речи // Изв. Томск. политехн. ун-та. 2009. Т. 315, № 5. С. 121.

9. *Азаров И. С., Вашкевич М. И., Лихачев Д. С., Петровский А. А.* Изменение частоты основного тона речевого сигнала на основе гармонической модели с нестационарными параметрами // Тр. СПИИРАН. 2014. Вып. 32. С. 5—26.

10. *Тиунов С. Д., Мещеряков Р. В., Черных Д. В.* Оптимизация вычисления одновременной маскировки речевого сигнала // Тр. СПИИРАН. 2014. Вып. 32. С. 45—57.

11. *Качковская Т. В.* Использование темпоральных характеристик для сегментации речевого потока на крупные смысловые единицы (на материале русского языка) // Тр. СПИИРАН. 2014. Вып. 32. С. 68—81.

12. *Басов О. О., Носов М. В., Шалагинов В. А.* Исследование характеристик джиттера периода основного тона речевого сигнала // Тр. СПИИРАН. 2014. Вып. 32. С. 27—44.

Сведения об авторах

Виктор Юрьевич Будков — канд. техн. наук; СПИИРАН, лаборатория речевых и многомодальных интерфейсов; научный сотрудник; E-mail: budkov@iias.spb.su

Андрей Леонидович Ронжин — д-р техн. наук, профессор; СПИИРАН, лаборатория речевых и многомодальных интерфейсов; зам. директора по научной работе; E-mail: ronzhin@iias.spb.su

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК. 519.711.72

А. Н. Павлов, Д. А. Павлов, Б. В. Москвин, К. Л. Григорьев

МОДИФИЦИРОВАННАЯ МОДЕЛЬ ГИБКОГО ПЕРЕРАСПРЕДЕЛЕНИЯ ТЕХНОЛОГИЧЕСКИХ ОПЕРАЦИЙ ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ

Представлена модификация математической модели процесса планирования децентрализованной обработки информации в динамически изменяющихся условиях с учетом временных ограничений на выполнение операций обработки, хранения и передачи информационных потоков в космических системах.

Ключевые слова: *динамическая сеть, информационное взаимодействие, неявные временные ограничения.*

Анализ основных тенденций развития современных информационных технологий показывает, что несмотря на значительный рост производительности аппаратно-программных средств и повышение интенсивности передачи потоков данных существенно возрастают требования потребителей к оперативности доставки информации, ее полноте и качеству, а также возникают новые более информационно-емкие задачи.

Эта проблема особенно актуальна для стремительно развивающегося космического сектора информационного обеспечения гражданских и военных потребителей, так как помимо известных проблем, имеющих место в любой информационно-вычислительной системе, на качество функционирования космических систем (КС) влияет ряд ограничений, а именно: динамическое изменение структуры информационного взаимодействия КС, обусловленное преимущественно баллистикой движения космических аппаратов (КА); особенности целевого функционирования КС; слабая пропускная способность каналов космической связи (по сравнению с пропускной способностью наземных каналов связи); функционирование КС в рамках жестких временных и ресурсных ограничений.

Существующие подходы к управлению информационными КС (ИКС) зачастую обращены к так называемым „слепым“ методам реконфигурации, как правило, сводящимся к имитационному моделированию разрабатываемых и известных систем, к рассмотрению частных сторон их функционирования и выявлению лишь общих характеристик. В современных условиях такой подход к управлению сложными ИКС представляется неперспективным, так как не позволяет ответить на главные вопросы — как оптимально (рационально) распределять информационные потоки; где, когда и сколько информации необходимо получать, хранить, обрабатывать и отправлять потребителю в динамически изменяющихся условиях и при жестких временных ограничениях.

В качестве примера ИКС может быть рассмотрена система, представляющая собой информационную сеть КА и обеспечивающая обработку и информационное взаимодействие КА

друг с другом и с наземными потребителями информации. Структура и параметры такой сети из-за постоянного орбитального движения КА и ряда других ограничений изменяются, или, другими словами, в пространстве формируется динамическая сеть (ДС). При этом предполагается, что структура и параметры (характеристики) сети изменяются в дискретные моменты времени, когда весь временной интервал (интервал планирования) разбивается на подынтервалы, соответствующие постоянству структуры. Также предполагается, что в самом общем случае каждый элемент (узел) ДС оборудован унифицированной многофункциональной аппаратурой и может выполнять такие технологические операции, как хранение, передача и обработка потоков информации различного вида; кроме того, известны технические характеристики указанной аппаратуры — объем запоминающего устройства (ЗУ) в каждом узле, производительность вычислительных средств; пропускная способность каналов связи между узлами сети. Логическая взаимосвязь технологических операций, выполняемых в узлах ДС, представлена схемой, приведенной на рис. 1.

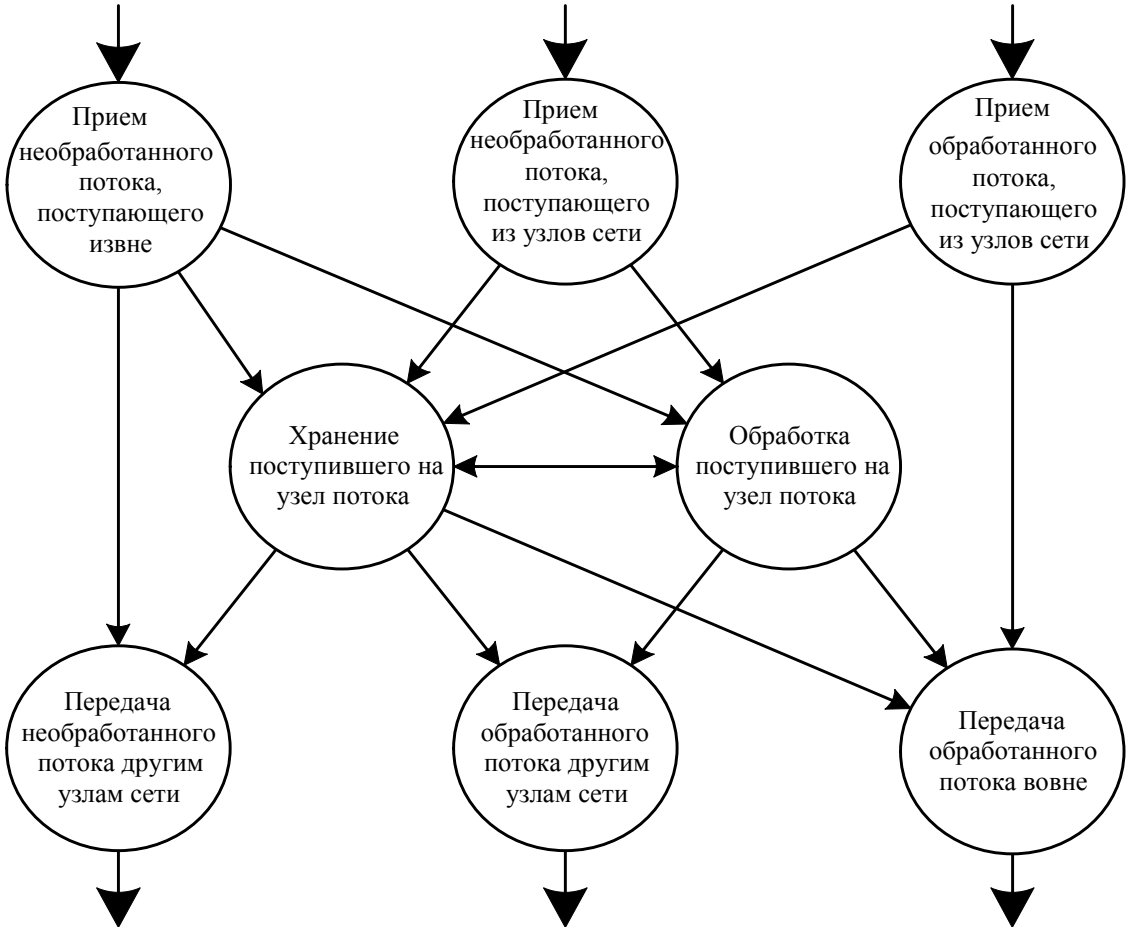


Рис. 1

При таком подходе к рассмотрению ИКС особую актуальность приобретает постановка и решение задачи гибкого перераспределения технологических операций передачи, обработки и хранения многопотоковой информации в динамической сети с учетом временных ограничений, заданных интервалами постоянства ее структуры.

Рассмотрим сеть с двумя типами потоков — потоком целевой информации и потоком телеметрической информации (рис. 2). Треугольниками обозначены ЗУ узлов сети объемом V_i, $i=\overline{1,7}$, прямоугольниками — блоки обработки с интенсивностью $\kappa_{i\rho}, i=\overline{1,7}, \rho \in [p,c]$, потоков информации в узлах, также указаны интенсивности $\omega_{ij\rho}, i=\overline{1,7}, j=\overline{1,7}, i \neq j$, передачи того или иного вида информации между узлами сети. Потоки информации поступают через

узлы *1* и *6*, узел *4* является центральным узлом распределения обработанной информации, узел *5* характеризует потребителя информации. Поступающая информация, с интенсивностью $\psi_{i\rho}$, может обрабатываться как в узлах *1* и *6*, так и в промежуточных узлах *2*, *3* и *7*.

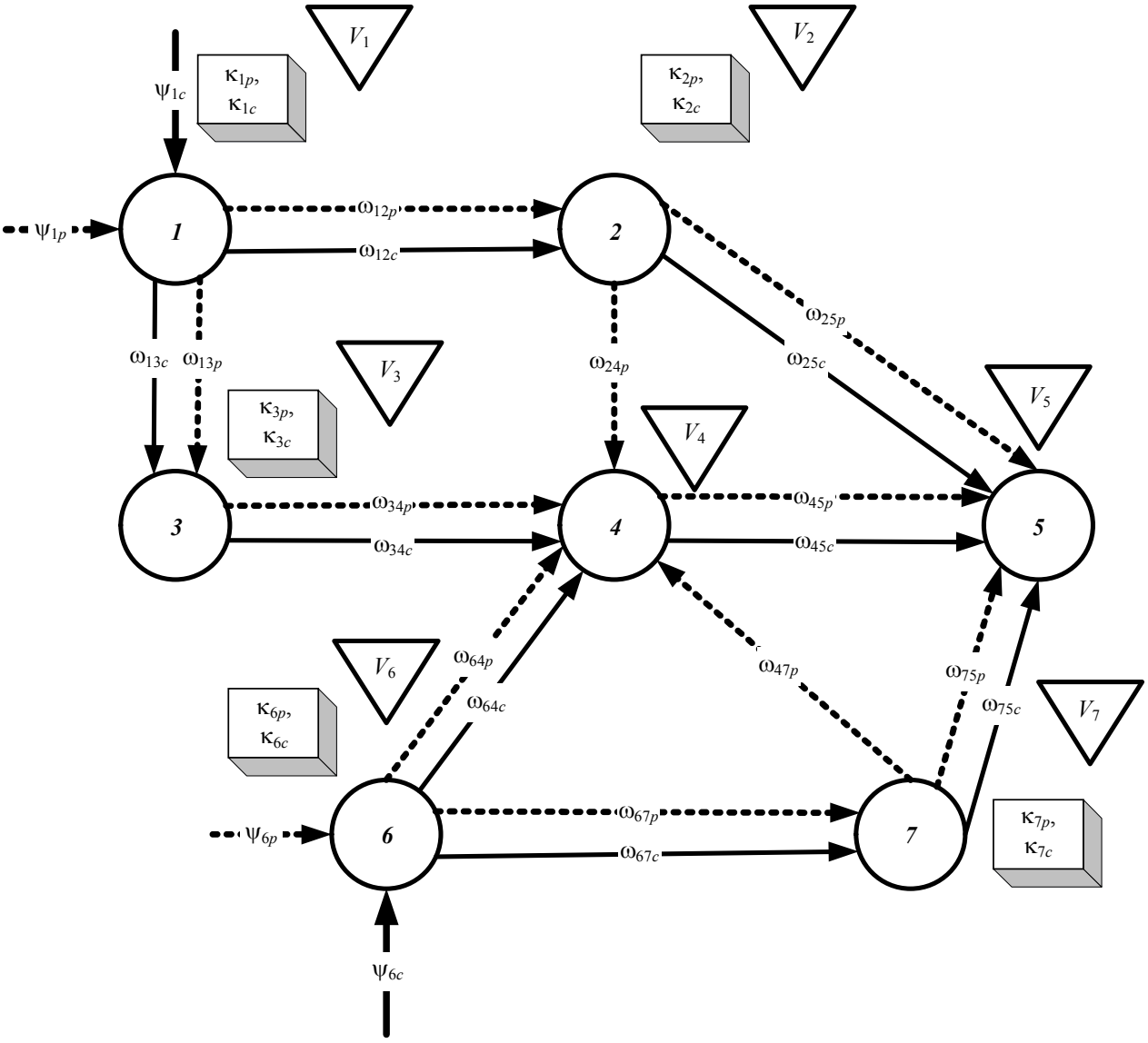


Рис. 2

Вариант структурной динамики рассматриваемой сети (в упрощенном виде) представлен на рис. 3.

Содержательная постановка задачи планирования гибкого перераспределения технологических операций информационного взаимодействия элементов ДС имеет следующие особенности:

— если объемы поступающей информации превышают возможности ДС по обработке, хранению и передаче данных, то невостребованная информация теряется;

— необходимо учитывать, что суммарное время последовательного выполнения технологических операций обработки и передачи потоков информации ограничивается длительностью интервала постоянства структуры (T_k);

— необходимо учитывать энергетические расходы на выполнение указанных технологических операций.

В целом задача состоит в нахождении плана по обработке, хранению и передаче информационных потоков в целях обеспечения потребителей полной и качественной информацией с минимальными энергетическими затратами.

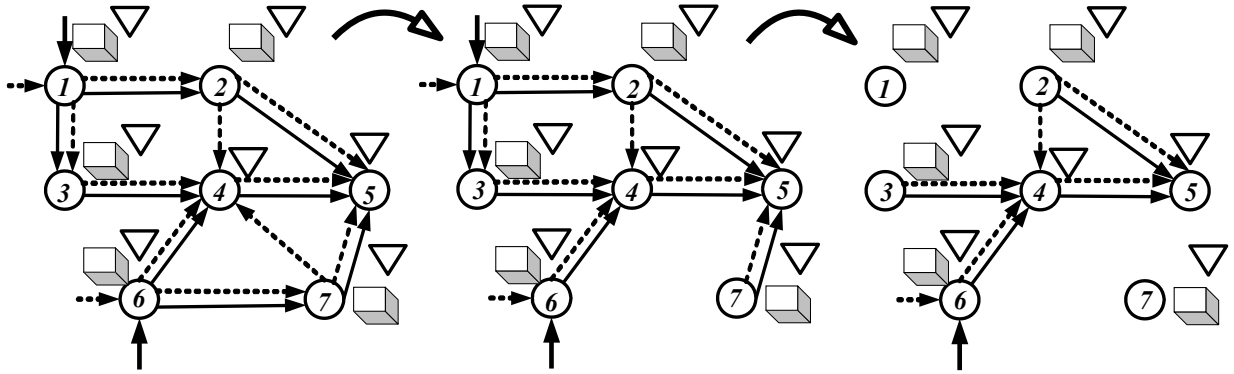


Рис. 3

В работах [1, 2] приведена математическая модель процесса планирования гибкого перераспределения операций управления потоком между элементами и подсистемами информационной системы. Данная модель позволяет определить агрегированный вариант технологии приема, хранения и обработки данных для χ-й структуры динамической сети:

$$\alpha_3\alpha_2\sum_{\rho=1}^{p}\lambda_\rho\sum_{i=1}^{n_\chi}\sum_{k=1}^{L_\chi}g_{\chi i\rho k}-\alpha_3\alpha_1\sum_{\rho=1}^{p}\gamma_\rho\sum_{i=1}^{n_\chi}\sum_{k=1}^{L_\chi}z_{\chi i\rho k}-\alpha_4\upsilon_\chi\to\max; \tag{1}$$

$$\left(\sum_{j\in N_{\chi i}^{+}}x_{\chi ij\rho k}-\sum_{j\in N_{\chi i}^{-}}x_{\chi ji\rho k}\right)+\left(y_{\chi i\rho k}-y_{\chi i\rho(k-1)}\right)+g_{\chi i\rho k}+z_{\chi i\rho k}=I_{\chi i\rho k},$$
$$i\in N_\chi,\ \rho\in \mathrm{P},\ k=1,...,L_\chi; \tag{2}$$

$$\upsilon_\chi-\sum_{\rho=1}^{p}\sum_{j=1}^{n_\chi}r_{ij\rho}\sum_{k=1}^{L_\chi}x_{\chi ij\rho k}-\sum_{\rho=1}^{p}\beta_{i\rho}\sum_{k=1}^{L_\chi}g_{\chi i\rho k}-v_{\chi i}=R_{\chi 0i},\ i\in N_\chi; \tag{3}$$

$$y_{\chi i\rho k}+\eta_{\chi ik}=V_{\chi i},\ i\in N_\chi,\ \rho\in \mathrm{P},\ k=1,...,L_\chi; \tag{4}$$

$$0\le x_{\chi ij\rho k}\le\omega_{\chi ij\rho k}(t_k-t_{k-1}),\ 0\le g_{\chi i\rho k}\le\psi_{\chi i\rho k}(t_k-t_{k-1}),\ i\in N_\chi,\ \rho\in \mathrm{P},\ k=1,...,L_\chi; \tag{5}$$

$$y_{\chi i\rho k}\ge 0,\ z_{\chi i\rho k}\ge 0,\ v_{\chi i}\ge 0,\ \eta_{\chi ik}\ge 0,\ \upsilon_\chi\ge 0,\ i\in N_\chi,\rho\in \mathrm{P},\ k=1,...,L_\chi. \tag{6}$$

В выражениях (1)—(4) $\alpha_1,\alpha_2,\alpha_3,\alpha_4\ge 0$ $(\alpha_1+\alpha_2+\alpha_3+\alpha_4=1)$ — коэффициенты значимости заданных показателей, устанавливаемые лицом, принимающим решение, в конкретных условиях функционирования системы; λ_ρ — коэффициент сжатия информации после обработки; γ_ρ — коэффициент, характеризующий значимость теряемого потока ρ-го типа; переменные $x_{\chi ji\rho k}$, $y_{\chi i\rho k}$, $y_{\chi i\rho(k-1)}$, $g_{\chi i\rho k}$, $z_{\chi i\rho k}$ являются неизвестными и характеризуют соответственно объем переданного потока ρ-го типа из i-го узла в j-й на интервале T_k, объем сохраненного потока ρ-го типа в i-м узле на интервале T_k, объем сохраненного потока ρ-го типа в i-м узле на интервале T_{k-1}, объем обработанного потока ρ-го типа в i-м узле на интервале T_k и объем потерянного потока ρ-го типа в i-м узле на интервале T_k; $r_{ij\rho}$ — коэффициент энергетических затрат на передачу информации от i-го узла j-му; $\beta_{i\rho}$ — коэффициент энергетических затрат на обработку потока ρ-го типа; $N_{\chi i}^{+}$ — множество узлов, способных передать информацию i-му узлу; $N_{\chi i}^{-}$ — множество узлов, способных принять информацию от i-го узла.

Выражения (5) соответствуют ограничению времени, отведенному на выполнение технологических операций в рамках текущего интервала T_k (t_{k-1}, t_k — начало и окончание интервала); здесь величина $\omega_{\chi ij\rho k}$ характеризует интенсивность передачи потока ρ-го типа из *i*-го узла в *j*-й, а величина $\psi_{\chi i\rho k}$ — интенсивность обработки потока ρ-го типа в *i*-м узле.

Для того чтобы учесть указанную выше особенность рассматриваемой задачи, связанную с длительностью интервала постоянства структуры, требуется ввести в математическую модель (1)—(6) дополнительные переменные и ограничения.

Так, введем новые переменные, отражающие длительность выполнения операций передачи и обработки информации, тогда объемы информационных потоков можно представить следующим образом:

$$x_{\chi ij\rho k} = \omega_{\chi ij\rho k} t^x_{\chi ij\rho k}, \;\; g_{\chi i\rho k} = \psi_{\chi i\rho k} t^g_{\chi i\rho k}, \;\; i, j \in N_\chi, \;\; \rho \in \mathrm{P}, \;\; k = 1,..., L_\chi, \tag{7}$$

где $t^x_{\chi ij\rho k}$ — искомое время, требуемое на передачу потока объемом $x_{\chi ij\rho k}$; $t^g_{\chi i\rho k}$ — время, требуемое на обработку потока объемом $g_{\chi i\rho k}$.

Тогда временные ограничения на обработку всех информационных потоков в *i*-м узле и передачу из *i*-го узла *j*-му определяются выражениями

$$0 \le \sum_{\rho=1}^{p} t^g_{\chi i\rho k} \le t_k - t_{k-1} = T_k, \;\; 0 \le \sum_{\rho=1}^{p} t^x_{\chi ij\rho k} \le t_k - t_{k-1} = T_k, \;\; i \in N_\chi, \rho \in \mathrm{P}, \;\; k = 1,..., L_\chi. \tag{8}$$

Наряду с введенными переменными необходимо ввести ограничения на суммарное время последовательной обработки и передачи информационного потока ρ-го типа от источников до потребителей. В рассматриваемом случае имеет смысл осуществлять поиск всех путей прохождения информационного потока ρ-го типа от каждого его источника до каждого конечного получателя, тогда в общем виде временные ограничения можно записать следующим образом:

$$0 \le \sum_{a=1}^{N_w} t^g_{\chi i_a\rho k} + \sum_{a=2}^{N_w} t^x_{\chi i_{(a-1)} i_a\rho k} \le t_k - t_{k-1} = T_k, \;\; w \in W^k, \;\; \rho \in \mathrm{P}, \;\; k = 1,..., L_\chi, \tag{9}$$

где $w \in W^k$ — конкретный путь $w = <(i_1, i_2), (i_2, i_3), ..., (i_{N_w-1}, i_{N_w})>$ из множества всех рассматриваемых путей прохождения информационного потока ρ-го типа на *k*-м интервале времени.

Таким образом, посредством замены переменных в модели (1)—(6) по формулам (7) и ввода ограничений (8), (9) вместо ограничений (5) получена модифицированная математическая модель планирования гибкого перераспределения технологических операций информационного взаимодействия элементов ИКС.

В заключение отметим следующее. Неявные временные ограничения принято относить к классу косвенных ограничений, учет которых в „высоконагруженных“ мобильных информационных системах, к которым относятся ИКС, является гарантией правильного планирования комплекса операций информационного взаимодействия. В целом, как показывают предварительные эксперименты, основное достоинство разработанной модели состоит в том, что по сравнению с ранее предложенными моделями и алгоритмами планирования, ориентированными на определение агрегированного варианта гибкого перераспределения операций приема, хранения и обработки данных, ее применение позволяет повысить качество планов передачи и обработки информации в динамически изменяющихся условиях.

Исследования, выполненные по данной тематике, проводились при финансовой поддержке ведущих университетов Российской Федерации: Санкт-Петербургского государственного политехнического университета (мероприятие 6.1.1), Университета ИТМО (субсидия 074–U01), Программы научно-технического сотрудничества Союзного государства „Мониторинг СГ“

(проект 1.4.1–1), Российского фонда фундаментальных исследований (гранты № 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120, 13-06-0087), Программы фундаментальных исследований ОНИТ РАН (проект № 2.11), проектов ESTLATRUS 2.1/ELRI-184/2011/14, 1.2/ELRI-121/2011/13.

СПИСОК ЛИТЕРАТУРЫ

1. Комбинированные модели управления структурной динамикой сложных технических объектов / *Б. В. Москвин, Е. П. Михайлов, А. Н. Павлов, Б. В. Соколов* // Изв. вузов. Приборостроение. 2006. Т. 49, № 11. С. 8—12.

2. *Павлов А. Н.* Комплексное моделирование структурно-функциональной реконфигурации сложных объектов // Тр. СПИИРАН. 2013. Вып. 5. С. 143—168.

Сведения об авторах

Александр Николаевич Павлов — канд. техн. наук, доцент; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; E-mail: pavlov62@list.ru

Дмитрий Александрович Павлов — адъюнкт; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург; E-mail: dpavlov239@mail.ru

Борис Владимирович Москвин — канд. техн. наук, профессор; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург; E-mail: z-moskvin@mail.ru

Кирилл Леонидович Григорьев — канд. техн. наук; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург; E-mail: Grigorjev.kir@yandex.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 629.7.06.062

А. Ю. КУЛАКОВ

МОДЕЛЬ ОЦЕНИВАНИЯ РАСХОДА ТОПЛИВА КОСМИЧЕСКОГО АППАРАТА С УЧЕТОМ НЕШТАТНЫХ СИТУАЦИЙ

Предложена упрощенная модель процесса функционирования космического аппарата, предназначенная для оценочного расчета его топливного ресурса и срока активного существования. При построении учитывается влияние сбоев (отказов) бортовой аппаратуры системы управления движением на расход топлива как при штатной работе, так и в нештатных ситуациях.

Ключевые слова: *моделирование сбоев и отказов, топливный ресурс, нештатные ситуации, бортовые системы КА.*

Одним из основных показателей функционирования космического аппарата (КА) на рабочей орбите является срок его активного существования, который жестко регламентирует пределы невосстанавливаемых ресурсов бортовой аппаратуры (БА) и запас топлива в баках двигательной установки (ДУ) КА. При этом запас топлива в баках на фиксированный срок зависит от периодичности включения установки и единичного расхода топлива при различных режимах работы КА. Поэтому для штатной работы номинальное значение топливного ресурса определяется числом режимов функционирования ДУ КА в заданном интервале времени (сутки, месяц, год). К ним относятся режимы коррекций орбиты, которые задаются ис-

ходя из известных параметров орбиты КА, и режимы стабилизации, применяемые для сброса кинетического момента или погашения угловых скоростей КА при отделении его от ракеты-носителя. Однако при летной эксплуатации возможны незапланированные режимы работы двигательной установки, вызванные нештатным функционированием БА, в частности сбоями и отказами приборов системы управления движением.

Сбои и отказы носят вероятностно-периодический характер [1]. Факторы, определяющие нештатную работу приборов системы управления движением, можно разделить на две группы: технологические, связанные с особенностью работы прибора, и физические, связанные с воздействием космической среды на КА. При построении математической модели учет этих факторов посредством уравнений является сложной задачей, а воспроизведение строгой зависимости отказов от данных факторов (в ходе летной эксплуатации КА) практически невозможно. Поэтому для расчета расхода топлива предлагается воспользоваться логико-вероятностными методами учета периодичности сбоев.

При моделировании нештатных ситуаций (НС) для оценивания ресурсов КА возникают трудности, связанные с неопределенностью состояния моделируемого объекта. Моделирование же штатных режимов выполняется при строго заданных условиях эксплуатации. Штатные режимы заранее определяются техническим описанием систем КА, где они уже заложены в логику бортового программного обеспечения (БПО). Для получения временных зависимостей параметров работы двигательной установки в штатных режимах обычно осуществляется аппроксимация результатов типовых расчетов для начального, конечного и промежуточных состояний КА (или используется аналитическая зависимость этих параметров). Ситуации, вызванные сбоями БА, которые не определяются техническим описанием (нештатные ситуации), а являются известными по опыту эксплуатации аналогичных образцов техники, также могут быть промоделированы. При этом условия возникновения и протекания различных режимов могут быть описаны с помощью вероятностных зависимостей, что составляет дополнительную трудность при оценке топливного ресурса, а следовательно, и срока активного существования КА.

Представим модель КА как динамическую систему, характеризуемую базовыми множествами: множеством моментов времени $T = \{t\}$, множеством входных ситуаций $Q = \{q\}$ и множеством состояний системы $X = \{x\}$. Множество входных ситуаций характеризуется вектором $\boldsymbol{\theta}_q = \{\theta_1, \theta_2, ..., \theta_i, ..., \theta_n\}$, где $\theta_i \in \{0; 1\}$, n — количество диагностируемых бортовых элементов. Множество состояний системы определяется остатками топлива и совокупностью работоспособных приборов.

Пусть множество вариантов работы ДУ КА формализовано как для штатных режимов, так и для нештатных ситуаций. Каждый элемент этого множества определяется двумя параметрами: частотой возникновения ω и расходом топлива ρ. Запишем эти параметры в виде функций:

$$\omega = \begin{cases} \omega(t) \text{ для штатных режимов,} \\ \omega(\boldsymbol{\theta}(t)) \text{ для нештатных ситуаций;} \end{cases}$$

$$\rho = \begin{cases} \rho(M(t)) \text{ для штатных режимов,} \\ \rho(M(t), \boldsymbol{\theta}(t)) \text{ для нештатных ситуаций,} \end{cases}$$

где функция $M(t)$, характеризуемая зависимостью массовых и центровочных характеристик от времени, может задаваться линейной или экспоненциальной зависимостью, например при аппроксимации экспериментальных результатов; в нештатных ситуациях на параметры ω и ρ влияют случайные величины — время сбоя бортовых элементов и время их восстановления.

Для имитации сбоев и отказов приборов системы управления движением предлагается использовать наиболее распространенный на практике экспоненциальный закон распределения, который характерен для периода штатной работы системы, исключая наработку и старение БА.

Этот закон задается интенсивностью безотказной работы и наиболее вероятно описывает внезапные сбои. Таким образом, время внезапного отказа вычисляется по формуле [2]

$$t_{\text{отк}} = -\frac{\ln \xi}{\lambda},$$

где λ — интенсивность отказов, ξ — случайное число с равномерным законом распределения на интервале (0, 1].

В качестве исходных данных для определения интенсивности отказов используются статистические данные по эксплуатирующимся аналогичным приборам: если статистические данные отсутствуют, то расчет интенсивности осуществляется на основе зависимости

$$\lambda = -\frac{\ln P(\tau)}{\tau},$$

где $P(\tau)$ — вероятность безотказной работы в течение временного интервала τ.

При имитации сбоев учитывается также время восстановления работоспособности системы: $T_{\text{в}} = T_{\max}\xi$, где $T_{\max}$ — максимальное время восстановления, определяемое особенностью работы бортовой системы. Тогда вектор $\boldsymbol{\theta}_q(t)$ в каждый момент времени будет задаваться следующим образом:

$$\theta_i(t) = \begin{cases} 1 & \text{при } t \in (t_{\text{отк}}; t_{\text{отк}} + T_{\text{в}}), \\ 0 & \text{при } t \notin (t_{\text{отк}}; t_{\text{отк}} + T_{\text{в}}). \end{cases}$$

Зависимость работоспособности системы от возникшей НС описывается событиями, инициирующими отказ. Сбои и отказы приборов системы являются такими событиями. Дерево отказов (сбоев) системы управления движением при возникновении нештатных ситуаций, сопровождаемых дополнительным расходом топлива, представлено на рис. 1, где приняты следующие обозначения: СГ — силовой гироскоп, ОУ — оконечное устройство, БУП — блок управления приводом, БПМ — блок питания мотора, ЗД — звездный датчик, ИУС — измеритель угловой скорости; показаны элементарные отказы и составные (выделены фоном).

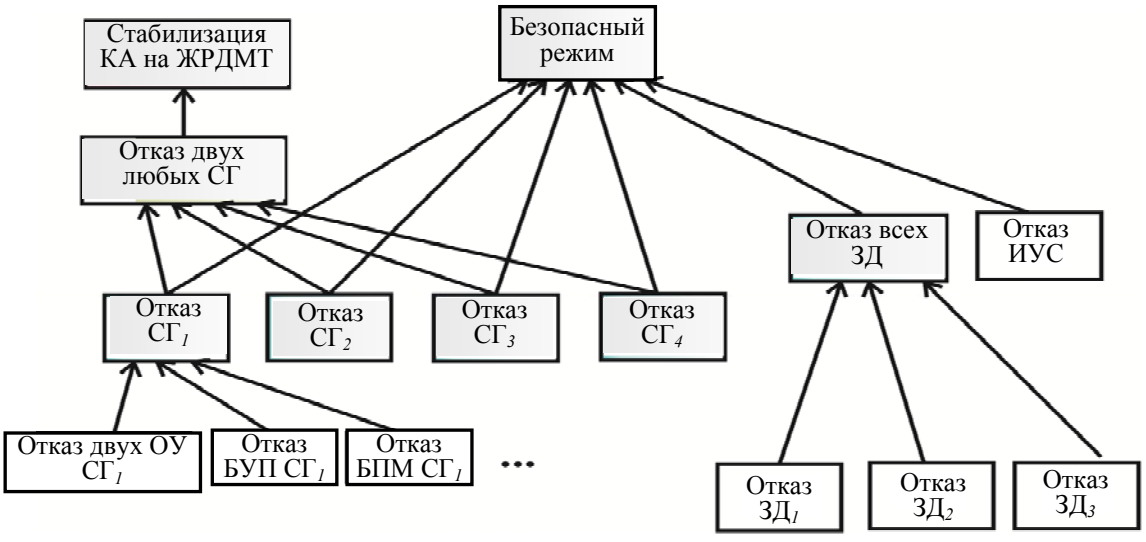


Рис. 1

Для оценочного расчета расхода топлива с учетом нештатных ситуаций воспользуемся аппаратом логико-вероятностного исчисления [3].

Рассмотрим два варианта НС — уход КА в безопасный режим (БР) с последующим выводом с помощью двигательной установки и стабилизацию КА на жидкостных реактивных двигателях малой тяги (ЖРДМТ). Опишем данные режимы с помощью логических функций, построенных в соответствии с рис. 1:

$$f_{\text{СГ}_i} = F_{\text{СГ}_i}\left(\theta_{\text{ОУ}_1\text{СГ}_i}, \theta_{\text{ОУ}_2\text{СГ}_i}, \theta_{\text{БУП СГ}_i}, \theta_{\text{БМП СГ}_i}\right);$$

$$f_{\text{БР}} = F_{\text{БР}}\left(f_{\text{СГ}_1}, f_{\text{СГ}_2}, f_{\text{СГ}_3}, f_{\text{СГ}_4}, \theta_{\text{ЗД}_1}, \theta_{\text{ЗД}_2}, \theta_{\text{ЗД}_3}, \theta_{\text{ИУС}}\right);$$

$$f_{\text{ЖРДМТ}} = F_{\text{ЖРДМТ}}\left(f_{\text{СГ}_1}, f_{\text{СГ}_2}, f_{\text{СГ}_3}, f_{\text{СГ}_4}\right),$$

где $F_{\text{СГ}_i}$ — условие сбоя *i*-го силового гироскопа, $F_{\text{БР}}$ — условие ухода КА в безопасный режим; $F_{\text{ЖРДМТ}}$ — условие стабилизации КА на ЖРДМТ.

При штатном режиме эксплуатации КА функция $\omega(t) = f(t)$ задается следующим образом:

$$\omega(t) = \begin{cases} 1 \text{ при } t > T; \\ 0, \end{cases}$$

где T — текущий момент включения двигательной установки, вычисляемый в зависимости от времени предыдущего аналогичного включения.

Тогда в общем виде интенсивность расхода топлива при работе ДУ КА в каждый момент времени определяется формулой

$$y(t) = \sum_{j=1}^{m} \omega_j(\boldsymbol{\theta}(t)) \cdot \rho_j(M(t), \boldsymbol{\theta}(t)),$$

где m — общее число возможных вариантов работы установки.

Для определения расхода топлива в различных режимах был использован программный комплекс математического моделирования динамики КА и работы системы управления движением [4]. Были проимитированы варианты работы ДУ КА в штатных режимах и нештатных ситуациях.

В данном комплексе КА рассматривается как материальный объект, имеющий сложную конструкцию с гибкими выносимыми элементами, а моделируемая система управления включает чувствительные элементы и исполнительные органы системы управления движением, а также БПО, определяющее логику функционирования КА. Пространственное движение КА описывается тремя моделями — углового движения, движения центра масс по орбите и колебания гибких элементов конструкции. Учитывается влияние на работу КА внутренних и внешних сил и моментов. Схематичное описание программного комплекса представлено на рис. 2 в виде блочной структуры.

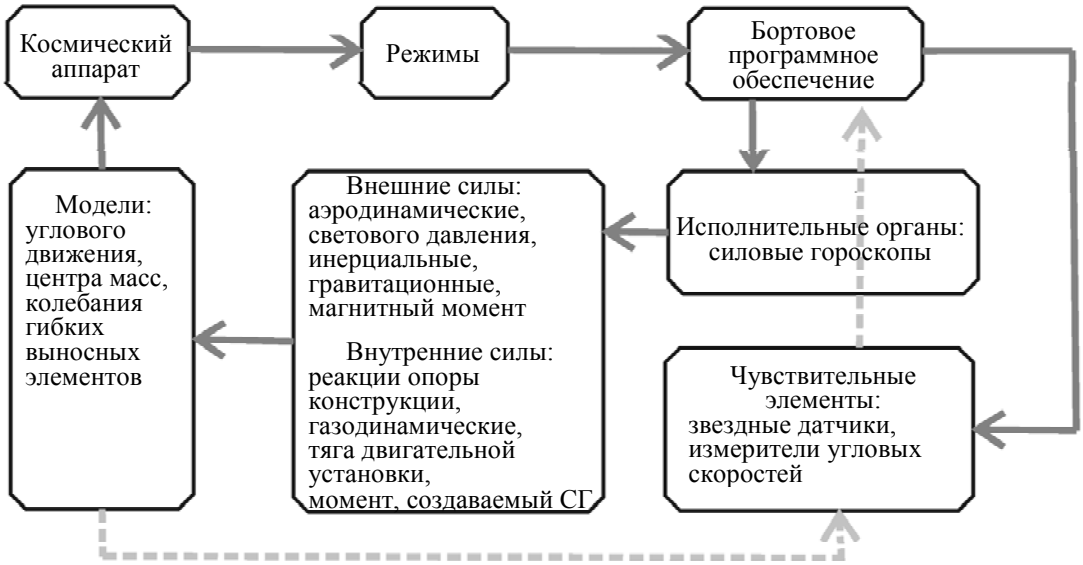


Рис. 2

Результаты проведенных экспериментов при среднем времени между сбоями $t_{\text{ср}}$=20…40 суток представлены в таблице, где приведены: масса остатка топлива, приходящегося

на момент фиксированного окончания срока активного существования КА (M_1); суммарная масса топлива, необходимого для вывода КА из безопасного режима (M_2); масса топлива, необходимого для стабилизации КА на ЖРДМТ (M_3), и возможный срок активного существования (D), при определении которого учитывается также запас топлива на увод КА с орбиты. Интенсивность сбоев, определяемая как величина, обратная $t_{ср}$, рассчитывается для являющихся причинами отказов элементов силового гироскопа и чувствительных элементов.

λ, $ч^{-1}$					M_1, кг	M_2, кг	M_3, кг	D, лет
ОУ	БУП	БПМ	ЗД	ИУС				
0,001	0,001	0,001	0,001	0,001	324,3	81,5	10,5	7,2
0,002	0,001	0,002	0,002	0,001	279,8	106,5	46,5	6,49
0,002	0,002	0,002	0,002	0,002	257,5	114,4	63	6

Предложенная модель оценки срока активного существования КА позволяет учитывать внезапно возникающие сбои и отказы бортовой аппаратуры, что обеспечивает приближение оценки расхода основного невосполняемого ресурса (топлива двигательной установки) к более реальному прогнозу. Представленная логико-вероятностная модель может быть расширена и применительно к другим системам, обеспечивающим функционирование КА.

Исследования, выполненные по данной тематике, проводились при финансовой поддержке ведущих университетов Российской Федерации: Санкт-Петербургского государственного политехнического университета (мероприятие 6.1.1), Университета ИТМО (субсидия 074–U01), Программы научно-технического сотрудничества Союзного государства „Мониторинг СГ“ (проект 1.4.1–1), Российского фонда фундаментальных исследований (гранты № 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120, 13-06-0087), Программы фундаментальных исследований ОНИТ РАН (проект № 2.11), проектов ESTLATRUS 2.1/ELRI-184/2011/14, 1.2/ELRI-121/2011/13.

СПИСОК ЛИТЕРАТУРЫ

1. *Базовский И.* Надежность. Теория и практика. М.: Мир, 1965. 373 с.

2. *Лохматкин В. В., Куренко В. И.* Прогнозирование производительности съемки КА ДЗЗ с учетом надежности бортовых систем // Изв. Самарского науч. центра РАН. 2013. Т. 15, № 4 (2).

3. *Рябинин И. А.* Надежность и безопасность структурно-сложных систем. СПб: Политехника, 2000. 320 с.

4. *Сотников М. В., Копылов В. М., Игнатьев М. Г.* Программный комплекс динамического моделирования работы СУД КА „ММДКА“ // Тр. III науч.-техн. конф. молодых ученых и специалистов ФГУП «КБ „Арсенал“». СПб: 2011.

Сведения об авторе

Александр Юрьевич Кулаков — аспирант; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; E-mail: russ69@bk.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 004.05

В. В. Бураков

МОДЕЛИРОВАНИЕ И ИДЕНТИФИКАЦИЯ ДЕФЕКТОВ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММНОГО КОДА

Представлен подход к моделированию и идентификации дефектов программного кода для улучшения качества программного обеспечения. Подход базируется на графовом моделировании исходного кода приложения и его комплексном анализе.

Ключевые слова: *моделирование программ, графовая модель, дефект кода.*

Введение. При создании программного обеспечения бортовых комплексов широко используется объектно-ориентированное визуальное моделирование — бурно развивающаяся в настоящее время область компьютерной инженерии. В начале 1990-х гг. по этой теме появилось много фундаментальных работ. Наибольшее влияние на формирование этой области оказали исследования Г. Буча, И. Джакобсона, Д. Рамбо, П. Коуда, Д. Харела, Б. Селика и др., усилиями которых был создан стандарт в этой отрасли — язык унифицированного моделирования UML (Unified Modeling Language) [1].

На сегодняшний день UML является широко используемым средством для проектирования программных комплексов любой сложности. Несмотря на это, UML имеет ряд существенных недостатков.

— *Неточная семантика*. Так как UML определяется комбинацией собственных атрибутов (абстрактный синтаксис), языка объектных ограничений — OCL (Object Constraint Language) (формальная проверка правильности) и естественного английского языка (подробная семантика), то он лишен согласованности, присущей языкам, точно определенным техниками формального описания. В некоторых случаях абстрактный синтаксис UML, OCL и английский язык противоречат друг другу, в других случаях — не полностью соответствуют друг другу. Неточность описания самого UML одинаково отражается и на пользователях, и на разработчиках программных продуктов, что приводит к несовместимости инструментов из-за уникальной интерпретации спецификаций.

— *Отсутствие полноты по Тьюрингу*.

— *Кумулятивная нагрузка/рассогласование нагрузки*. Как и в любой системе обозначений, UML позволяет описывать одни системы более кратко и эффективно, чем другие. Таким образом, разработчик склоняется к решениям, которые обеспечивают сочетание „сильных“ сторон UML и языков программирования. Проблема становится более очевидной, если язык разработки не соответствует традиционным принципам.

Концепции UML не позволяют смоделировать готовую версию программного продукта, поэтому достаточно сложно выявить ошибки в архитектуре приложения на стадии проектирования. Помимо этого, в течение жизненного цикла программного продукта необходимо учитывать накладные расходы к общей сумме трудозатрат на разработку, связанные с необходимостью поддерживать модель в согласованном с кодом состоянии. Эти факторы могут крайне негативно повлиять на надежность программного обеспечения, что недопустимо в сферах, где от работы аппаратно-программного комплекса может зависеть здоровье и жизнь человека. Учитывая тенденцию всесторонней компьютеризации жизнедеятельности человека, надежность кода можно считать одним из основополагающих критериев качества. Из-за большого объема и высокого уровня сложности программного кода наиболее эффективным

способом выявления дефектов в системах подобного класса является автоматизированный поиск.

В этой связи особую актуальность приобретает возможность строгого и согласованного в математическом смысле моделирования структуры и поведения разрабатываемых программных систем, а также создания моделей и алгоритмов для поиска дефектов архитектуры на любом этапе жизненного цикла программного обеспечения. Предлагаемый в настоящей статье подход основан на использовании теории графов для представления программного кода и включает набор методов и алгоритмов идентификации дефектов.

Графовая модель кода. Основой предлагаемой концепции является моделирование программного кода с помощью ориентированного помеченного типизированного графа. Помимо основного графа, представляющего исходный код программы, используется множество подграфов, описывающих дефекты и условия их появления. Для более полного описания условий идентификации дефектов программного кода введены допустимые и недопустимые графы [2].

Определение 1. *Ориентированный граф* $G = (V, E, s, t)$ состоит из двух множеств — конечного множества V, элементы которого называются вершинами, и конечного множества E, элементы которого называются дугами. Каждая дуга связана с упорядоченной парой вершин. Для обозначения вершин используются символы $v_1, v_2, v_3\ldots$, а для обозначения дуг — символы $e_1, e_2, e_3\ldots$ Если $e_1 = (v_i, v_j)$, то v_i — начальная вершина дуги e_1, а v_j — ее конечная вершина. Все дуги, имеющие одну пару начальных и конечных вершин, называются параллельными. Функции $s: E \to V$ и $t: E \to V$ связывают с каждой дугой одну начальную и одну конечную вершины.

Определение 2. *Помеченный граф.* Пусть $L=(VL, EL)$, $A=(VA)$ — пара непересекающихся потенциально бесконечных множеств меток и ролей соответственно. (L, A)-помеченный граф G представляет собой тройку (g, l, a), такую что: $g=(V, E, s, t)$ — граф; $l=(vl: V\to VL, el: E\to EL)$ — заданная пара функций соответственно вершин и дуг, при этом функция vl является инъективной; $a=(va: V\to VA)$ — функция отображения вершин на множество ролей.

Определение 3. *Помеченный типизированный граф.* Пусть $T=(VT, ET)$ — пара непересекающихся конечных множеств предопределенных типов вершин и дуг. (L, A)-помеченный T-типизированный граф G представляет собой двойку $(g, type)$, такую что: g — (L, A)-помеченный граф; $type=(vt: V\to VT, et: E\to ET)$ — пара функций, где функция vt связывает с каждой вершиной из множества V ее тип из множества VT, а функция et связывает с каждой дугой из множества E ее тип из множества ET.

Определение 4. *Подграф.* H является подграфом G (обозначается $H\subseteq G$), если существует инъективный графовый морфизм $m: H\to G$, называемый соответствием H в G.

Определение 5. *Допустимый (недопустимый) граф.* Пусть $T=(VT, ET)$ — пара непересекающихся конечных множеств типов вершин и дуг, а TT — T-помеченный граф, называемый допустимым (недопустимым) графом. (L, A)-помеченный TT-типизированный граф представляет собой двойку (G, tt), такую что: G — (L, A)-помеченный граф; $tt: G\to TT$ — тотальный LGraph-морфизм. Помеченный TT-типизированный графовый морфизм представляет собой графовый морфизм вида $f: G\to H$ между (L, A)-помеченными TT-типизированными графами. TT-сохраняющий помеченный типизированный графовый морфизм представляет собой помеченный TT-типизированный графовый морфизм вида $f: G\to H$, такой что $tt_H: f=tt_G$ для $\forall x\in \mathrm{dom}(f)$.

На рис. 1 показан пример представления исходного кода программы на языке C++ в виде графовой модели.

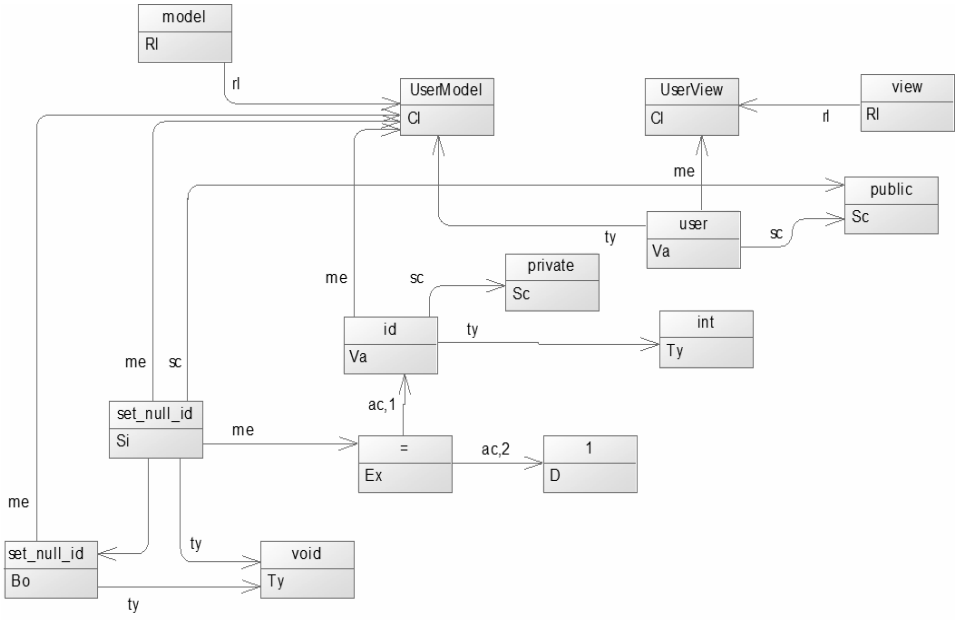


Рис. 1

Ниже представлен исходный код на языке C++ для данного графа:

```
class UserView {//role = view
  public: UserModel user;};
class UserModel{ //role = model
                private:
                        int id;
        public:
                        void set_null_id(); };
public: void UserModel::set_null_id(){
                         id = 0;}
```

Спецификация и поиск дефектов. В основе предлагаемого подхода лежит представление программного кода и описание дефекта в виде графовой модели. В простейшем случае задача выявления дефекта сводится к поиску подграфа в графе. В терминах графовой модели дефектом может являться как факт нахождения подграфа в графе, так и отсутствие искомого подграфа.

Помимо поиска подграфа в графе, реализация предлагаемого подхода предусматривает:

— распознавание дефектов на основе метрического анализа;

— использование ролей для специализации контекста разрабатываемых программных сущностей и поиска дефектных программных решений;

— спецификацию условий и поиск программных сущностей, нарушающих эти условия.

Конструктивно эти варианты поиска дефектов оформлены в виде плагинов, которые можно комбинировать в зависимости от конкретных задач анализа кода.

Метрический анализ. Суть метрического анализа заключается в подсчете количественных характеристик кода. По умолчанию могут быть определены основные базовые метрики:

— количество входных вершин вершины;

— количество выходных вершин вершины;

— количество входных вершин для вершины с дугами определенного типа;

— количество выходных вершин для вершины с дугами определенного типа;

— длина нисходящего пути вершины из дуг определенного типа.

На основе базовых метрик пользователь может сформировать производные метрики. Каждая производная метрика определяется путем введения функциональной зависимости от

n (*n*>0) других как базовых, так и производных метрик. Основными видами функциональных зависимостей, порождающих производные метрики, являются:

— сумма *n* базовых (производных) метрик;

— частное от деления одной базовой (производной) метрики на другую;

— максимальное значение базовой (производной) метрики;

— минимальное значение базовой (производной) метрики;

— среднее (арифметическое, геометрическое и т.п.) значение базовых (производных) метрик.

Для каждой метрики возможно задать граничные значения, что позволяет более гибко настраивать систему под различные проекты.

Роли и контекст. Каждому классу в проекте присваивается роль (или роли), определяющая контекст использования экземпляров этого класса другими программными элементами. Также задаются правила отношений между классами с определенными ролями. В каждом правиле есть возможность указать как разрешенные связи, так и запрещенные. На основе заданных правил для идентификации дефекта производится поиск запрещенных связей между сущностями программного кода. Например, пусть необходимо выявить корректность реализации архитектуры „модель—представление—контроллер“, которая заключается в отделении логики модели от логики представления кода и контроллера, т.е. необходимо исключить прямые обращения из класса модели в классы представления и контроллера. На рис. 2 показаны графы, моделирующие эти дефекты. Суть идентификации дефекта заключается в нахождении подграфов в графе исходного кода.

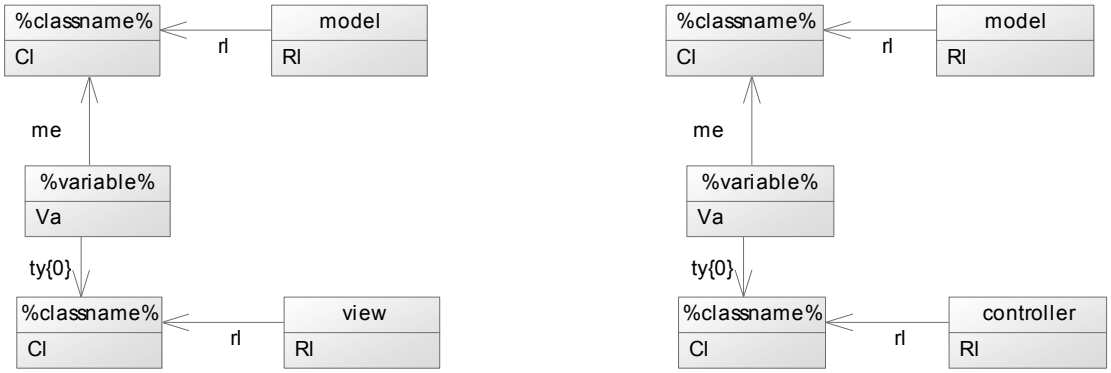


Рис. 2

Условия. В практике программирования относительно редко встречаются случаи, когда какая-либо часть программного кода является сама по себе дефектом. Как правило, для этого необходимо выполнение одного или нескольких условий.

Каждый элемент списка условий — есть граф или набор графов, представляющих некое условие. Каждый элемент имеет свой порядковый номер, необходимый для случаев, когда важен порядок выполнения условий. Если порядок выполнения условий важен только для нескольких элементов списка, то порядковые номера задаются только для них. Помимо этого, каждому элементу присваивается признак, содержащий информацию о том, является данное условие обязательным или нет.

Граф, описывающий некое условие, также может быть допустимым или недопустимым. Для выявления некоторого дефекта необходимо, чтобы все обязательные условия были выполнены. Необязательные условия являются уточнениями к выявлению дефекта, так как не всегда могут существовать в исходном коде.

Практическая апробация. Для практической оценки эффективности предлагаемого подхода была разработана программная система. В качестве платформы для ее реализации выбрана среда разработки MS Visual Studio. Компоненты, реализующие вышеуказанные

этапы процессов моделирования и поиска дефектов, представляют собой набор программных модулей, включенных в плагин к этой среде (рис. 3).

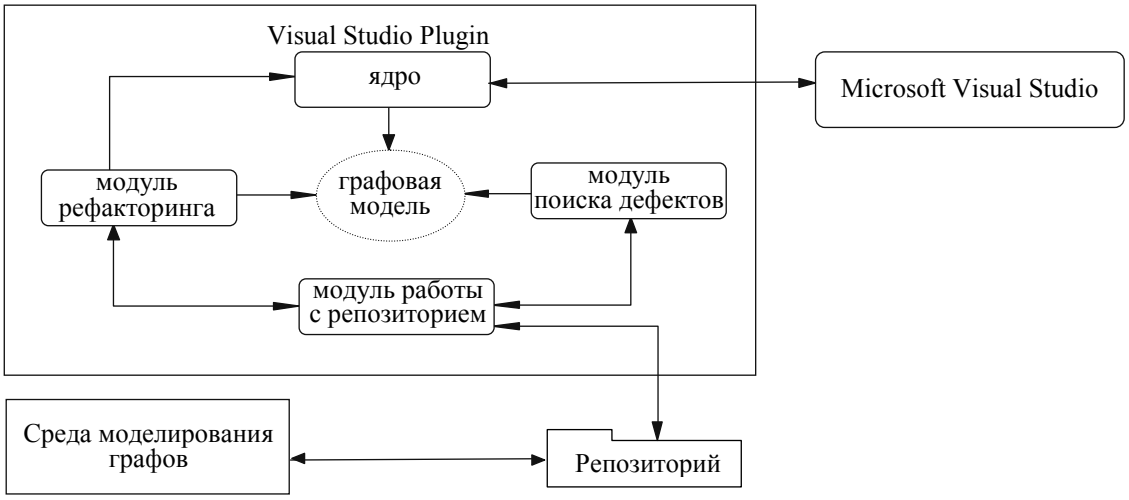


Рис. 3

Среда взаимодействует непосредственно с модулем „ядро“, который позволяет работать с объектной моделью открытого проекта. Большая часть модуля „ядро“ представляет собой парсер, преобразующий код проекта в графовую модель. Непосредственно в сам плагин не входят репозиторий дефектов и рефакторингов и среда моделирования графов. Репозиторий представляет собой совокупность каталогов на диске, содержащих XML-файлы с описанием того или иного дефекта или рефакторинга, описанного в виде графа. Модуль поиска дефектов, получив описание дефектов из репозитория, производит поиск в графовой модели, сгенерированной при помощи модуля „ядро“.

Основные возможности разработанной системы:

— математически точное описание структуры и поведения классов;

— математическое моделирование дефектов;

— возможность ведения и пополнения базы дефектов;

— использование разных стратегий для поиска дефектов, выбор наиболее подходящей из них, возможность комбинирования стратегий;

— гибкая настройка системы поиска;

— автоматизированный поиск дефектов.

Заключение. В настоящее время осуществляется опытная эксплуатация разработанной системы моделирования и поиска программных дефектов и формирование статистической информации, на основе которой можно будет оценить эффективность подхода.

Предложенный подход позволит автоматизировать процессы выявления дефектов в архитектуре программных приложений и обеспечить повышение их качества и снижение затрат на разработку и сопровождение.

Исследования, выполненные по данной тематике, проводились при финансовой поддержке ведущих университетов Российской Федерации: Санкт-Петербургского государственного политехнического университета (мероприятие 6.1.1), Университета ИТМО (субсидия 074–U01), Программы научно-технического сотрудничества Союзного государства „Мониторинг СГ“ (проект 1.4.1–1), Российского фонда фундаментальных исследований (гранты № 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120, 13-06-0087), Программы фундаментальных исследований ОНИТ РАН (проект № 2.11), проектов ESTLATRUS 2.1/ELRI-184/2011/14, 1.2/ELRI-121/2011/13.

СПИСОК ЛИТЕРАТУРЫ

1. OMG UML Version 2.3 [Электронный ресурс]: <http://www.omg.org/spec/UML/2.3/>, 2010.

2. *Бураков В. В.* Управление качеством программных средств. СПб: СПбГУАП, 2009. 287 с.

Сведения об авторе

Вадим Витальевич Бураков — д-р техн. наук, профессор; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; E-mail: Burakov@eureca.ru

Рекомендована СПИИРАН

Поступила в редакцию 10.06.14 г.

УДК 681.3.062

Л. Н. ФЕДОРЧЕНКО

МЕТОД РЕГУЛЯРИЗАЦИИ ГРАММАТИК В СИСТЕМАХ ПОСТРОЕНИЯ ЯЗЫКОВЫХ ПРОЦЕССОРОВ

Описывается алгоритм регуляризации приведенных контекстно-свободных грамматик, основанный на эквивалентных преобразованиях, который совместно с алгоритмом устранения рекурсий редуцирует грамматику к единственному регулярному выражению.

Ключевые слова: *контекстно-свободная грамматика, эквивалентное преобразование грамматики.*

Регулярные множества и контекстно-свободные языки с различными ограничениями и расширениями успешно применяются в технологии построения трансляторов. При создании разного вида трансляторов языков программирования используется множество технологических средств построения анализаторов формальных языков. Как правило, эти технологические средства обеспечивают лишь проверку предъявляемых к грамматике требований и выдачу диагностических сообщений об их нарушениях. Для получения эквивалентной грамматики, удовлетворяющей условиям алгоритма анализа, существуют способы эквивалентных преобразований контекстно-свободных (КС) грамматик. Эти преобразования могут быть выполнены автоматически. Такие эквивалентные преобразования реализованы в программном средстве SynGT (Syntax Graph Transformations), разработанном в СПИИРАН.

В настоящей статье рассматривается алгоритм редукции приведенной КС-грамматики к одному регулярному выражению с помощью эквивалентных преобразований.

Определим отношение R зависимости между нетерминалами КС-грамматики следующим образом.

О п р е д е л е н и е 1. Пусть $G = (V_N, V_T, P, S)$ — КС-грамматика, где V_N — алфавит нетерминалов, V_T — алфавит терминалов, P — множество правил грамматики, $S \in V_N$ — начальный символ грамматики. Будем считать, что нетерминал $A \in V_N$ зависит от нетерминала $B \in V_N$, если существует правило вида $A \to \alpha B \beta \in P$, где $\alpha,\ \beta \in V^*,\ V = V_N \cup V_T$. Этот факт будем записывать как $(A, B) \in \mathbb{D}$, а множество всех таких пар $\mathbb{D}$ будем называть *отношением зависимости между нетерминалами* КС-грамматики. Другими словами, $\mathbb{D} \subseteq V_N \times V_N$. При $A = B$ считаем нетерминал A *самозависимым (рекурсивным).*

Определение 2. Нетерминал $A \in V_N$ назовем *абсолютно независимым*, если $\neg\exists B : (A, B) \in \mathbb{D}$.

Другими словами, абсолютно независимые нетерминалы определяются правилами, в правых частях которых нет ни одного нетерминала.

Рассмотрим схему эквивалентного преобразования КС-грамматики в одно регулярное выражение.

1. Множество всех нетерминалов V_N данной приведенной КС-грамматики разбивается на непересекающиеся подмножества (уровни) l_i, $0 \le i \le k < |N|$, в соответствии с уровнем зависимости нетерминала. Нетерминалы, правила для которых содержат только терминальные символы в правых частях, принадлежат к самому нижнему нулевому уровню l_0, т.е. такие нетерминалы $A \in V_N$, значения (регулярные множества) $R(A)$ которых содержатся в правой части соответствующего A-правила с терминальными символами.

2. Для каждого следующего уровня l_i и для всех нетерминалов этого уровня осуществляется замещение (подстановка) нетерминалов уровня l_{i-1} их значениями. Самый последний уровень l_k содержит только один элемент — начальный символ S грамматики, который замещается регулярным выражением на последнем шаге преобразования.

Все подстановки осуществляются в соответствии с иерархией на нетерминалах, определяемой отношением зависимости $\mathbb{D}$, т.е. если правая часть A_j-правила содержит вхождение нетерминала B_j, то пара (A_j, B_j) принадлежит отношению зависимости $\mathbb{D}$ согласно определению 1.

Задача состоит в том, чтобы разбить все нерекурсивные нетерминалы на непересекающие множества s_0, s_1, s_2, ... , s_m , $m \le n$, где $n = |V_N|$, которые должны обладать следующими свойствами.

1) Все нетерминалы $A \in s_0$ абсолютно независимы, т.е. регулярные выражения для таких нетерминалов *априори* определены A-правилами.

2) Нетерминалы любого множества s_i, $1 \le i \le m$, *независимы* друг от друга, т.е. для любой пары нетерминалов $(A, B) \in s_i$ выполняется условие $(A, B) \notin \mathbb{D}$.

3) Нетерминалы множества s_i, $1 \le i \le m$, *непосредственно* вычислимы по регулярным значениям нетерминалов из множеств s_j, $1 \le j \le i-1$, предыдущих уровней. Другими словами, если $A \in s_i$, то в правой части A-правила все вхождения нетерминалов замещаются регулярными значениями вычисленных к этому моменту нетерминалов из множества s_j, $1 \le j \le i-1$.

Примем, что нетерминалы из множества s_l относятся к уровню l. Очевидно, что на максимальном уровне (m) всегда располагается начальный нетерминал S грамматики, и только он, если он не встречается в правых частях правил. Это условие задается предварительно [1].

На уровне „0“ находятся нетерминалы, регулярные значения которых уже определены: их можно назвать опорными. Далее, при параллельном продвижении по уровням, „вычисляются“ регулярные значения всех других нетерминалов путем замещения в соответствии с вышеуказанным свойством 3. В результате такого процесса формируется регулярное выражение для начального нетерминала S грамматики на уровне m. Именно оно и есть искомый результат эквивалентных преобразований исходной КС-грамматики.

Лево- и/или *правосторонние* и *центральная* рекурсии в преобразуемых правилах, в случае их обнаружения, исключаются согласно приведенному в работах [2, 3] алгоритму с использованием операций итерации (бинарной или унарной).

В качестве примера будем использовать грамматику с трактовкой правил как регулярных формул с операциями объединения, конкатенации и обобщенной итерации [3]. В процессе преобразований правил грамматики могут появляться и другие регулярные операции: два вида замыкания — *рефлексивно-транзитивное*, обозначаемое „звездочкой“ Клини, и *транзитивное*, обозначаемое „плюсом“ Клини, а также обобщенная итерация, обозначаемая как „#“ (см. определение в работах [2, 3]).

Пример 1. Дана КС-грамматика $G = (V_N, V_T, P, S)$, где

$$V_T = \{'d', '.', '\backslash', 'e', '+' ; '-'\};$$

$$V_N = \{ A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}, A_{12}, A_{13}, A_{14}, A_{15}\}; S = A_{15};$$

P состоит из 15 правил.

Список правил грамматики приведен на рис. 1.

A_1: '+' ; '–'.	A_2: A_1 ; ε.	A_3: A_2 , A_{14}.	A_4: '\' ; 'e'.
A_5: A_4 , A_3.	A_6: A_{14} ; A_{11}.	A_7: A_6 , A_5.	A_8: '.' , A_{14}.
A_9: A_{14}.	A_{10}: A_9 ; ε.	A_{11}: A_{10} , A_8.	A_{12}: 'd'.
A_{13}: A_{12} ; A_{13} , A_{12}.	A_{14}: A_{13}.	A_{15}: A_{14} ; A_{11} ; A_7.	

Рис. 1

Альтернативы для каждого нетерминала A представлены в виде регулярных выражений с помощью операции объединения и разделены металингвистическом символом „ ; “, а конкатенируемые символы правых частей — символом „ , “; конец A-правила отмечен как „ . “. Таким образом, рассматривается КС-грамматика в регулярной форме, все правила которой имеют следующий вид:

<нетерминал>:<регулярное выражение>.

Строгие определения для обобщенных регулярных выражений в правилах КС-грамматик в регулярной форме приведены в работах [2, 3].

Отношение зависимости между нетерминалами $\mathbb{D}$ легко построить непосредственно по правилам грамматики в виде множества пар $\mathbb{D} \subseteq V_N \times V_N$. В результате получаем

$$\mathbb{D} = \{(A_2, A_1), (A_3, A_2), (A_3, A_{14}), (A_5, A_3), (A_5, A_4), (A_6, A_{11}), (A_6, A_{14}),$$
$$(A_7, A_5), (A_7, A_6), (A_8, A_{14}), (A_9, A_{14}), (A_{10}, A_9), (A_{11}, A_8), (A_{11}, A_{10}), (A_{13}, A_{12}),$$
$$(A_{13}, A_{13}), (A_{14}, A_{13}), (A_{15}, A_7), (A_{15}, A_{11}), (A_{15}, A_{14})\}.$$

Для наглядности отношение зависимости $\mathbb{D}$ можно представить и в виде матрицы (рис. 2). Строки и столбцы этой матрицы — нетерминальные символы грамматики. Непустой элемент матрицы, находящийся i-й строке и j-м столбце, помеченный символом T, означает, что нетерминал A_i зависит от нетерминала A_j (например, A_2 зависит от A_1, A_3 зависит от A_2 и A_{14} и т.д.). Пустые строки матрицы (например, 1, 4, 12-я) означают абсолютную, т. е. полную независимость нетерминалов A_1, A_4, A_{12} от всех других нетерминалов грамматики. Элемент $(A_{13}, A_{13})=T$ означает самозависимость нетерминала A_{13}, т. е. нетерминал A_{13} леворекурсивен. Применяя эквивалентное преобразование, получаем $A_{13} : A_{12}, (A_{12})^*. \equiv (A_{12})\#$.

Рассмотрим метод сортировки нетерминалов. Опишем в абстрактных терминах алгоритм сортировки нетерминалов КС-грамматики, исходя из построенного отношения зависимости. Цель алгоритма — расположить множество нетерминалов грамматики по уровням

таким образом, чтобы все нетерминалы одного уровня l, $1 \le l \le m$, зависели только от нетерминалов уровня $k < l$ в соответствии с отношением $\mathbb{D}$.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}
A_1															
A_2	T														
A_3		T												T	
A_4															
A_5			T	T											
A_6											T				
A_7					T	T									
A_8														T	
A_9														T	
A_{10}									T						
A_{11}								T		T					
A_{12}															
A_{13}												T	T		
A_{14}													T		
A_{15}							T				T			T	

Рис. 2

Алгоритм: *сортировка нетерминалов грамматики.*

Входные данные: $G = (V_N, V_T, P, S)$ — приведенная КС-грамматика, не содержащая рекурсивных нетерминалов; $\mathbb{D} \subseteq V_N \times V_N$ — отношение зависимости нетерминалов.

Выходные данные: $N = \{s_0, s_1, s_2, \ldots, s_m\}$, где $s_k \subseteq V_N$ — подмножество нетерминалов данной грамматики уровня k, $0 \le k \le m$.

Шаг 0. Расположение на уровне $l = 0$ всех абсолютно независимых нетерминалов:

$$l = 0;\ s_0 = \{A \mid \forall\, (A, B \in V_N)\colon (\neg\exists\, (\alpha, \beta \in V^*)\colon A \to \alpha B \beta \in P)\}.$$

Шаг 1. Построение множества нетерминалов следующего уровня:

$$l = l + 1;\ \ s_l = \{A \mid \forall\, (A \in V_N)\colon (\exists\, B \in V_N)\colon (B \in s_{l-1})\ \&\ (A, B) \in R\ \&\ (A \ne B)\}.$$

Шаг 2. Исключение из всех подмножеств s_k, $0 \le k \le l$–1, нетерминалов, содержащихся в подмножестве s_l:

```
for ∀ (A ∈ s_l):
do for k from 0 to l–1
     do if  A ∈ s_k  then  s_k := s_k \ {A} od
do;
```

Шаг 3. Определение необходимости продолжения процесса сортировки нетерминалов:

```
if s_l ≠ ∅ then goto Шаг 1;
```

Шаг 4. Окончание процесса сортировки:

$$m = l\text{–}1;\ \{\text{максимальный уровень нетерминалов}\}.$$

Результат: $N = \{s_0, s_1, s_2, \ldots, s_m\}$.

Применительно к рассмотренному примеру получен следующий результат:

$$s_0 = \{A_1, A_4, A_{12}\};\ s_1 = \{A_2, A_{13}\};\ s_2 = \{A_{14}\};\ s_3 = \{A_3, A_8, A_9\};\ s_4 = \{A_5, A_{10}\};$$

$$s_5 = \{A_{11}\};\ \ s_6 = \{A_6\};\ s_7 = \{A_7\};\ s_8 = \{\, A_{15}\}.$$

Динамика вычисления уровней зависимости показана на рис. 3, где символ T означает рассматриваемый нетерминал, а символ $\mathcal{T}$ — удаление его с более низкого уровня отношения

зависимости. Максимальный номер уровня $m = 8$. Заметим, что условие $A \neq B$, используемое на шаге 1 алгоритма, существенно: оно предотвращает бесконечный рост уровней.

	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}	A_{11}	A_{12}	A_{13}	A_{14}	A_{15}
8															T
7							T								$\not T$
6						T	$\not T$								$\not T$
5						$\not T$	$\not T$				T				$\not T$
4					T		$\not T$			T	$\not T$				
3			T		$\not T$	$\not T$		T	T						$\not T$
2			$\not T$				$\not T$							T	
1		T			$\not T$								T		
0	T			T								T			

Рис. 3

Используя результат сортировки нетерминалов и правила исходной КС-грамматики, построим регулярные выражения для всех нетерминалов, исходя из следующих соображений.

Регулярные значения нетерминалов, представленных регулярными выражениями (см. рис. 1), вычисляются на основе априорных значений тех правил грамматики, правые части которых являются регулярными выражениями над алфавитом терминалов. Такие априорные значения всегда существуют, поскольку грамматика приведенная. Все они относятся к нетерминалам уровня „0“.

В рассматриваемом примере три нетерминала уровня „0“: A_1, A_4 и A_{12}, и соответствующие регулярные выражения для них: $R(A_1) = (\text{'+'} ; \text{'–'})$, $R(A_4) = (\text{'\'} ; \text{'}e\text{'})$ и $R(A_{12}) = (\text{'}d\text{'})$.

Задание регулярных значений для всех нетерминалов уровня „0“ обеспечивает инициализацию процесса построения регулярных значений нетерминалов уровня l, $1 \le l$, по значениям нетерминалов предыдущих уровней. При этом можно утверждать, благодаря выполненной сортировке нетерминалов, что уже существуют регулярные значения, необходимые для вычисления значений нетерминалов уровня l путем замещения (подстановки) вхождений нетерминалов в правила КС-грамматики соответствующими регулярными выражениями, полученными на более низком уровне.

Согласно матрице зависимости (см. рис. 2), от регулярного значения нетерминала A_1 зависит значение нетерминала A_2, и только оно, и, следовательно, замещая (в соответствии с рис. 1) в правиле для A_2 вхождение A_1 значением ('+' ; '–'), получаем регулярное выражение (('+' ; '–'), ε) в качестве правой части нового правила для нетерминала A_2. Аналогичные рассуждения приводят к заключению, что регулярное значение $R(A_4) = (\text{'\'} ; \text{'}e\text{'})$ должно передаваться в правую часть правила для A_5, а $R(A_{12}) = (d)$ — для A_{13}. (Для упрощения записи исключим знаки-кавычки „ ' ' “ для обобщенного терминала d.) В результате подстановок получаем регулярные выражения $R(A_5) = ((\text{'\'} ; \text{'}e\text{'}), ((\text{'+'} ; \text{'–'}), \varepsilon), (d)+)$ и $R(A_{13}) = =((d), (d)^*)$ в качестве правых частей новых правил для нетерминалов A_5 и A_{13} уровня „1“.

Далее вычисляем значения нетерминалов уровня „2“, т.е. A_{14}. После подстановки в исходном правиле для A_{14} вместо A_{13} значения $R(A_{13})$ получаем $R(A_{14}) = (((d), (d)^*))$. Подобным же образом вычисляются регулярные значения нетерминалов следующих уровней вплоть до максимального, на котором находится всегда один — начальный — нетерминал грамматики $S = A_{15}$. Его регулярное значение и есть цель преобразований исходной КС-грамматики.

Итак, регуляризованная КС-грамматика G_1, эквивалентная исходной, — есть КС-грамматика в регулярной форме:

$$R(A_{15}) = R(S)\ ((d), (d)^*)\ ;$$
$$(((((d), (d)^*))\ ;\ \varepsilon), \text{'.'}, ((d), (d)^*))\ ;$$
$$(((d), (d)^*), ((((d), (d)^*))\ ;\ \varepsilon), \text{'.'}, ((d), (d)^*))\ ,$$

$$((\text{'\textbackslash'} ; \text{'}e\text{'}), (((((\text{'+'} ; \text{'–'}), \varepsilon), \varepsilon)), ((d), (d)^*)))) \equiv$$
$$\equiv d^+ ; d^*, \text{'.'}, d^+ ; (d^+; d^*, \text{'.'}, d^+), (\text{'\textbackslash'} ; \text{'}e\text{'}), [\text{'+'} ; \text{'–'}], d^+ \equiv$$
$$\equiv (d^+; d^*, \text{'.'}, d^+), [(\text{'\textbackslash'} ; \text{'}e\text{'}), [\text{'+'} ; \text{'–'}], d+].$$

Предложенный алгоритм получения регулярного выражения, эквивалентного приведенной КС-грамматике без ограничений на рекурсии, может быть применен при построении языковых процессоров. Используется метод сортировки нетерминалов КС-грамматики по отношению зависимости [4], реализованный в системе SynGT [2]. Приведенный алгоритм регуляризации позволяет обнаружить любые рекурсии (левая/правая/вложенная) по ходу распределения нетерминалов по уровням. Благодаря применению алгоритма удаления таких рекурсий [2, 3] снимаются ограничения на рекурсивности в исходной грамматике.

СПИСОК ЛИТЕРАТУРЫ

1. Handbook of Formal Languages / Eds.: *G. Rozenberg, A. Salomaa.* Berlin, Heidelberg, New York: Springer-Verlag, 1997. Vol. 2. 527 p.

2. *Fedorchenko L.* Regularization of Context-Free Grammars. Saarbrucken: LAP LAMBERT Academic Publishing, 2011.

3. *Федорченко Л. Н.* О регуляризации контекстно-свободных грамматик // Изв. вузов. Приборостроение. 2006. Т. 49, № 11. С. 50—54.

4. *Мартыненко Б. К.* Регулярные языки и КС-грамматики // Компьютерные инструменты в образовании. 2012. № 1. С. 14—20.

Сведения об авторе

Людмила Николаевна Федорченко — канд. техн. наук; СПИИРАН, ст. научный сотрудник;
E-mail: lnf@iias.spb.su

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

ТЕХНОЛОГИИ ПРОАКТИВНОГО УПРАВЛЕНИЯ СЛОЖНЫМИ ОБЪЕКТАМИ

УДК 519.8

С. А. Потрясаев

СИНТЕЗ СЦЕНАРИЕВ МОДЕЛИРОВАНИЯ СТРУКТУРНОЙ ДИНАМИКИ АСУ АКТИВНЫМИ ПОДВИЖНЫМИ ОБЪЕКТАМИ

Проанализированы возможные технологии синтеза сценариев моделирования структурной динамики АСУ активными подвижными объектами. Предложено при практической реализации сценариев использовать сервис-ориентированный подход, рассмотрена возможная структура соответствующего программно-математического обеспечения.

Ключевые слова: *комплексное моделирование, технологии синтеза сценариев моделирования, сервис-ориентированный подход.*

Введение. На базе концепции комплексного моделирования [1—2], которая, как показывает практика, может быть успешно реализована в рамках соответствующей имитационной системы, к настоящему времени решен широкий спектр задач анализа и синтеза автоматизированных систем управления (АСУ) группировками активных подвижных объектов (АПО). При исследовании различных прикладных задач важная роль в указанных системах отводится разработке специальных сценариев интерактивного взаимодействия лиц, принимающих решения, с имитационными системами. В настоящей статье рассматриваются особенности создания программно-математического обеспечения синтеза сценариев моделирования АСУ активными подвижными объектами на различных этапах их жизненного цикла.

Технологии синтеза сценариев моделирования структурной динамики АСУ АПО. Рассмотрим некоторые возможные сценарии моделирования процессов управления структурной динамикой АСУ АПО. Предположим, что обобщенное полимодальное многокритериальное описание задач управления структурной динамикой АСУ АПО имеет следующий вид [2—4]:

$$\mathbf{J}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\xi}(t), t) \to \underset{\mathbf{u}\in\Delta}{\text{extr}}; \tag{1}$$

$$\Delta = \Big\{\mathbf{u} \mid \mathbf{x}(t) = \boldsymbol{\varphi}\big(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\xi}(t), \boldsymbol{\beta}, t\big), \mathbf{y}(t) = \boldsymbol{\psi}\big(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\xi}(t), \boldsymbol{\beta}, t\big),$$
$$\mathbf{x}(T_0) \in X_0(\beta), \mathbf{x}(T_f) \in X_f(\beta), \mathbf{u}(t) \in Q\big(\mathbf{x}(t), t\big), \boldsymbol{\xi}(t) \in \Xi\big(\mathbf{x}(t), t\big), \beta \in B\mathbf{x}(t) \in \tilde{X}(t)\Big\}, \tag{2}$$

где $\mathbf{J}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\xi}(t), t) = \left\|\mathbf{J}_g^T, \mathbf{J}_o^T, \mathbf{J}_k^T, \mathbf{J}_p^T, \mathbf{J}_n^T, \mathbf{J}_e^T, \mathbf{J}_c^T, \mathbf{J}_v^T\right\|^T$ — вектор показателей эффективности функционирования АСУ АПО; $\mathbf{J}_g^T, \mathbf{J}_o^T, \mathbf{J}_k^T, \mathbf{J}_p^T, \mathbf{J}_n^T, \mathbf{J}_e^T, \mathbf{J}_c^T, \mathbf{J}_v^T$ — соответственно векторы

показателей эффективности управления движением, операциями взаимодействия, каналами, ресурсами, потоками, параметрами операции, структурами, вспомогательными операциями в АСУ АПО; индексы „*g*“, „*o*“, „*k*“, „*p*“, „*n*“, „*c*“, „*e*“, „*v*“ указанных векторов обозначают соответствующие модели управления (M_g, M_o, M_k, M_p, M_n, M_e, M_c, M_v); $\mathbf{x}(t)$, $\mathbf{y}(t)$ — соответственно обобщенные векторы состояния и выходных характеристик динамической системы, описывающей процессы управления структурной динамикой АСУ АПО; $\mathbf{u}(t)$ — обобщенный вектор управляющих воздействий, $\boldsymbol{\xi}(t)$ — обобщенный вектор возмущающих воздействий; $\boldsymbol{\beta}$ — вектор структурных параметров (характеристик) АСУ АПО, определяющих ее облик; $Q(\mathbf{x}(t),t)$, $\Xi(\mathbf{x}(t),t)$, B — заданные области изменений значений векторов $\mathbf{u}(t)$, $\boldsymbol{\xi}(t)$, $\boldsymbol{\beta}$; $X_0(\beta)$, $X_f(\beta)$, $\tilde{X}(t)$ — заданные области изменений значений вектора $\mathbf{x}(t)$ соответственно в начальный, конечный и текущий моменты времени.

В выражении (2) переходная и выходная функции $\boldsymbol{\varphi}\left(\mathbf{x}(t),\mathbf{u}(t),\boldsymbol{\xi}(t),\boldsymbol{\beta},t\right)$ и $\boldsymbol{\psi}\left(\mathbf{x}(t),\mathbf{u}(t),\boldsymbol{\xi}(t),\boldsymbol{\beta},t\right)$ в общем случае задаются в аналитико-алгоритмическом (имитационном) виде в рамках имитационной системы. Именно возможные варианты описания и реализации указанных функций (в более общем случае — операторов) определяют содержание методов и алгоритмов, которые могут быть положены в основу построения процедур получения скоординированных решений в задачах управления структурной динамикой АСУ АПО.

В рамках ранее выполненных исследований [2—4] указанные переходная и выходная функции задавались с использованием разработанного комплекса динамических моделей, а также соответствующего специального программного обеспечения, содержащего набор следующих программных модулей (ПМ):

„*Координация*“ — программный модуль многокритериального анализа и упорядочения вариантов функционирования АСУ АПО при различных сценариях изменения обстановки и внешних воздействий;

„*Надежность*“ — программный модуль расчета и многокритериального анализа показателей структурной надежности и устойчивости АСУ АПО;

„*Расписание*“ — программный модуль расчета плана функционирования наземного комплекса управления АПО, а также расчета показателей пропускной способности, оперативности и ресурсоемкости АСУ АПО для детерминированных сценариев изменения внешних воздействий;

„*Устойчивость*“ — программный модуль расчета и оптимизации показателей робастности и динамической устойчивости программ функционирования АСУ АПО для интервально заданных сценариев изменения внешних воздействий;

„*Пропускная способность*“ — программный модуль расчета показателей пропускной способности и ресурсоемкости АСУ АПО для стохастических сценариев изменения внешних воздействий;

„*Эффективность*“ — программный модуль расчета показателей эффективности функционирования АСУ АПО для стохастических сценариев изменения внешних воздействий.

Применительно к разработанной версии специального программного обеспечения в табл. 1 указаны (отмечены знаком „+“) классы моделей (аналитических или имитационных) АСУ АПО, которые были использованы для описания соответствующей подсистемы рассматриваемой АСУ.

Для многокритериального оценивания и упорядочения возможных сценариев функционирования АСУ АПО в штатных и заданных условиях ее применения разработана соответствующая методика формирования и расчета интегрального показателя качества и эффективности работы системы, базирующаяся на комбинированном использовании математического аппарата нечеткой логики и теории планирования эксперимента.

Таблица 1

Тип модели подсистемы АСУ АПО	Модели, реализованные в составе ПМ									
	„*Надежность*“		„*Расписание*“		„*Устойчивость*“		„*Пропускная способность*“		„*Эффективность*“	
	АМ	ИМ	АМ	ИМ	АМ	ИМ	АМ	ИМ	АМ	ИМ
АИМ тракта ТМИ	+	-	+	-	+	-	-	+	+	-
АИМ тракта ИТНП	+	+	+	-	+	-	-	+	+	-
АИМ тракта КПИ	+	-	+	-	+	-	-	+	+	-
АИМ тракта СИ	+	-	+	-	+	-	-	+	+	-
АИМ ЦУП АПО	+	-	+	-	+	+	-	+	+	-
АИМ СОД	+	-	+	-	+	-	-	+	+	-
АИМ внешней среды	+	+	-	-	+	+	-	+	+	-

Примечания: АИМ — аналитико-имитационная модель, АМ — аналитическая модель, ИМ — имитационная модель, ТМИ — телеметрическая информация, ИТНП — измерения текущих навигационных параметров, КПИ — командно-программная информация, СИ — специальная информация, ЦУП — центр управления полетом, СОД — система обмена данными.

К настоящему времени разработаны многочисленные подходы, способы, методы, алгоритмы и методики координационного анализа и выбора комплексов разнородных моделей, описывающих различные предметные области. В табл. 2 приведены возможные варианты взаимодействия частных программных модулей разработанного полимодельного комплекса для решения задач управления структурной динамикой АСУ АПО [2, 5].

Таблица 2

Процедура решения задачи	Модель задачи					
	$f_0^{(a)} \to \underset{\Delta^{(a)}}{\text{extr}}$	$f_0^{(a)} \to \underset{\Delta^{(u)}}{\text{extr}}$	$f_0^{(a)} \to \underset{\Delta^{(a)} \cap \Delta^{(u)}}{\text{extr}}$	$f_0^{(u)} \to \underset{\Delta^{(a)}}{\text{extr}}$	$f_0^{(u)} \to \underset{\Delta^{(u)}}{\text{extr}}$	$f_0^{(u)} \to \underset{\Delta^{(a)} \cap \Delta^{(u)}}{\text{extr}}$
АОМ→АР→К	+					
ИОМ→ АР →К				+	+	+
АОМ→ИОМ→ АР →К		+	+			
(АОМ⊂ИОМ)→ АР →К			+			
(ИОМ⊂АОМ)→ АР →К			+		+	+
$\begin{pmatrix} \text{АОМ}_1 \\ \cup \\ \text{ИОМ} \\ \cap \\ \text{АОМ}_2 \end{pmatrix}$ → АР →К				+	+	+

Примечания: АОМ — аналитическая оптимизационная модель; ИОМ — имитационная оптимизационная модель; АР — анализ полученных результатов (проводимый автоматически, либо с привлечением ЛПР); К — коррекция полученных решений; $\Delta^{(a)}$, $\Delta^{(u)}$ — множества (либо подмножества) допустимых альтернатив вида (2), заданных соответственно аналитически и алгоритмически; $f_0^{(a)}$, $f_0^{(u)}$ — обобщенные показатели эффективности функционирования АСУ АПО, заданные соответственно аналитически и алгоритмически [3—5].

Указанные в табл. 2 возможные схемы координации (согласования) моделей и показателей $f_0^{(a)}, f_0^{(u)}$ различаются способами генерации допустимых альтернативных решений; правилами проверки алгоритмически и аналитически заданных ограничений; способами перехода от одного шага интерактивного сужения множества допустимых альтернатив к другому шагу; алгоритмами взаимодействия с внешними системами (внешней средой), т.е. системами, не входящими в состав АСУ АПО, но оказывающими на нее влияние. Данное взаимодействие может иметь как индифферентный характер (например, объекты живой и неживой природы), так и целенаправленный.

В рамках существующих имитационных систем выбор той или иной схемы координации (см. табл. 2) осуществляется, как правило, в интерактивном режиме с обязательным участием соответствующих ЛПР. Для этого должны быть синтезированы возможные сценарии человеко-машинного взаимодействия ЛПР с имитационной системой, а также разработано соответствующее специальное программно-математическое обеспечение (СПМО). Остановимся на возможных подходах к решению задач формирования сценариев.

Программно-математическое обеспечение формирования сценариев моделирования. Предлагаемое модульное построение программно-математического обеспечения имитационных систем позволяет с помощью относительно небольшого количества вычислительных модулей построить множество вычислительных программ (совокупности вычислительных модулей) для моделирования как АПО, так и АСУ АПО в целом с различной степенью детализации в рамках одной и той же системы.

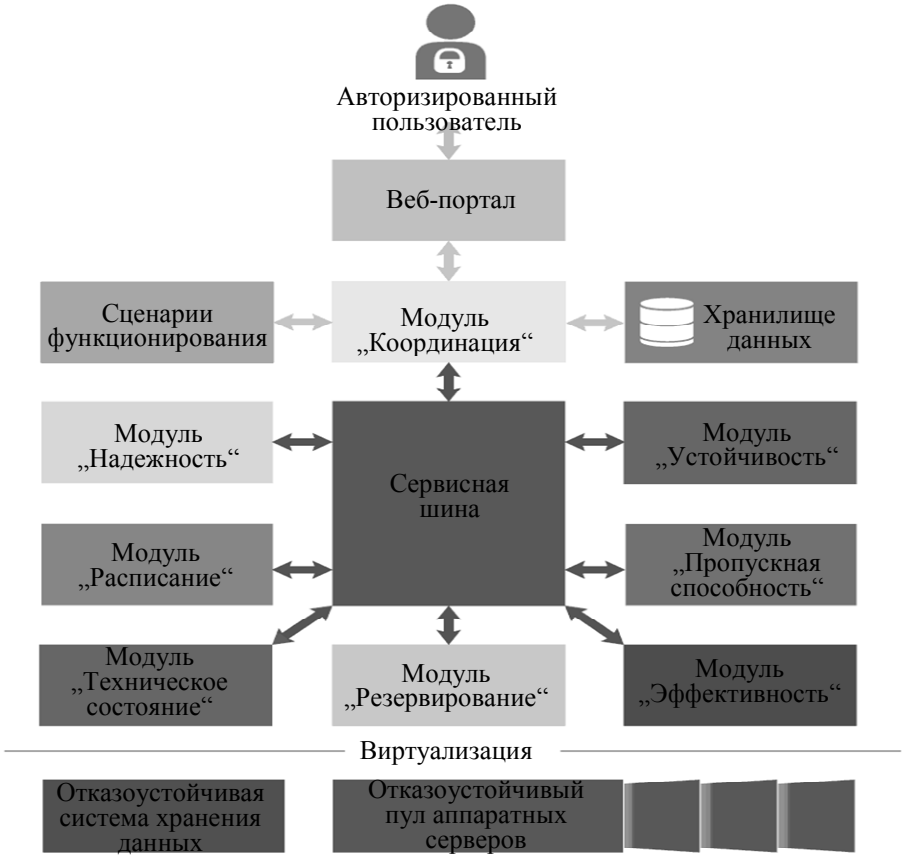
Как показывает предварительный анализ, на этапе практической реализации имитационной системы может возникнуть ряд трудностей, связанных, во-первых, с существенной гетерогенностью создаваемого СПМО и, во-вторых, с организацией взаимодействия между вычислительными модулями. Перечисленные выше программные модули будут содержать унаследованные программные подсистемы, представленные в виде законченных решений (например, стандартных процедур-функций), прошедших валидацию и верификацию. Такие подсистемы планируется использовать в ПМ „Надежность“, „Планирование“, „Эффективность“. Разработка подобных подсистем „с нуля“ представляет собой экономически невыгодный процесс как по трудозатратам, так и по времени выполнения проекта. В связи с этим необходимо в рамках имитационной системы синтезировать такие сценарии взаимодействия программных модулей, чтобы обеспечить между ними беспрепятственный обмен согласованными исходными данными и выходными результатами при решении различных классов задач анализа и синтеза АСУ АПО.

Наиболее хорошо зарекомендовавшим себя подходом к построению модульных систем на сегодняшний день является сервис-ориентированная архитектура (Service Oriented Architecture — SOA) [6—8]. Данный подход эффективен при слабой связности и распределенности используемых модулей, но требует в то же время использования стандартизированных интерфейсов модулей и работы по стандартизированным протоколам, что не было реализовано в унаследованных программных продуктах. При этом СПМО формирования сценариев, в терминах предлагаемой сервис-ориентированной архитектуры, соответствует программному обеспечению сервисной шины предприятия (Enterprise Service Bus — ESB) [6] и будет базироваться на свободно распространяемом программном каркасе Zato, который предоставляет разработчикам программного обеспечения возможность быстрого проектирования сервера приложений, реализующего сервисную шину предприятия [6—8].

С учетом указанных выше особенностей используемых унаследованных программных продуктов предлагается следующий вариант сервис-ориентированной архитектуры имитационной системы (см. рисунок).

Предлагаемая сервис-ориентированная архитектура позволяет перевести разрабатываемую программную имитационную систему в формат „облачного“ приложения, реализуемого

как сервис (Software as a Service — SaaS). Следствием перехода к облачным вычислениям является существенное повышение гибкости аппаратно-программной реализации. В частности, создаваемый программный комплекс может быть значительно распределен территориально и структурно, т.е. может быть реализован на базе вычислительных мощностей, принадлежащих разным организациям, в том числе находящимся в разных городах и странах. При этом синтезированная система, с точки зрения конечного пользователя, будет функционировать как целостное решение.



При объединении веб-сервисов в целях создания высокоуровневого приложения необходима стандартизация моделей их взаимодействия. Для стандартизации интерфейсов взаимодействия программных модулей, входящих в состав АСУ АПО, планируется использовать язык описания веб-сервисов WSDL (Web Services Description Language). Это язык, который описывает веб-сервис как набор конечных точек (портов), способных обмениваться сообщениями. WSDL служит для документирования и комплексного определения деталей взаимодействия распределенных систем в виде, удобном для машинной обработки.

Детальный анализ существующих технологий взаимодействия веб-сервисов показал, что при ориентации на интеграцию со сторонними системами наиболее строгим и перспективным для использования протоколом является SOAP.

Следует также заметить, что стандартизация моделей взаимодействия веб-сервисов не определяет логику работы программного комплекса. Для этих целей используются язык описания последовательности действий и инфраструктура для его реализации. В качестве такого языка целесообразно ориентироваться на стандарт моделирования последовательности взаимодействий веб-сервисов — язык Business Process Execution Language. В мае 2003 г. компании Microsoft, IBM, Siebel, BEA Systems и SAP совместно разработали первую версию спецификации этого языка для Web-Services (BPEL4WS или WS-BPEL) [9]. Указанная спецификация описывает язык, позволяющий моделировать поведение веб-сервисов при взаимодействии бизнес-процессов. Регламентируемый спецификацией механизм позволяет

координировать вычислительные процессы и компенсировать возникающие ошибки. Таким образом, рассмотренный выше язык WSDL определяет список возможных операций, а WS-BPEL задает порядок их выполнения. Спецификация поддерживает как структурные действия для управления потоком работ, так и базовые действия, которые включают взаимодействия с внешними веб-сервисами. Структурные действия определяют последовательность вызова веб-сервисов, а также поддерживают выполнение циклов и динамическое ветвление. По существу, они составляют основную логику программирования на WS-BPEL. Переменные используются для управления долговременным хранением данных в ходе обработки запросов веб-сервисов.

За последнее десятилетие WS-BPEL зарекомендовал себя как эффективный язык для описания логики работы приложений, основанных на распределенных веб-сервисах. Высокая частота его использования в современных распределенных приложениях, а также поддержка в различных программных средах исполнения позволяют обосновать целесообразность применения WS-BPEL в создаваемом программном комплексе [9].

Таким образом, синтез программной системы многокритериального оценивания и анализа показателей надежности и эффективности АСУ АПО на различных этапах жизненного цикла планируется осуществлять с помощью взаимодействия отдельных модулей программного комплекса на основе инструкций, описанных на языке WS-BPEL. При этом в дальнейшем планируется с использованием данного языка создать интеллектуальный интерфейс, базирующийся на результатах ранее выполненных исследований, основанных на динамической интерпретации процессов выполнения комплексов операций и синхронизированных с ними процессов получения/передачи, обработки, хранения и представления данных об АПО и АСУ АПО в целом. Использование интеллектуального интерфейса позволит пользователям не только описывать сценарии взаимодействия программных модулей, но и синтезировать данные сценарии, исходя из поставленной цели исследования. Указанные задачи могут быть решены в разрабатываемом ПМ „Координация“.

Заключение. Предложенная архитектура имитационной системы позволяет:

— для преодоления проблем гетерогенности, с одной стороны, и удобства развертывания системы, с другой, разместить все модули с несовместимыми требованиями к среде исполнения на различных виртуальных машинах в рамках одного аппаратного сервера;

— с учетом перспектив дальнейшего развития данных систем в направлении создания территориально-распределенных архитектур обеспечить взаимодействие модулей посредством сетевого обмена данными; для этих целей предлагается создать программные „обертки“, преобразующие частную систему ввода—вывода каждого модуля в стандартизированный интерфейс обмена данными;

— реализовать центральный модуль программного комплекса (ПМ „Координация“), позволяющий, в свою очередь, производить вызов всех остальных модулей, обеспечивать их согласованными исходными данными, осуществлять сбор и интерпретацию результатов;

— создать кроссплатформенный интегрированный пользовательский интерфейс, позволяющий удаленно использовать все возможности системы.

Исследования, выполненные по данной тематике, проводились при финансовой поддержке ведущих университетов Российской Федерации: Санкт-Петербургского государственного политехнического университета (мероприятие 6.1.1), Университета ИТМО (субсидия 074–U01), Программы научно-технического сотрудничества Союзного государства „Мониторинг СГ“ (проект 1.4.1–1), Российского фонда фундаментальных исследований (гранты № 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120, 13-06-0087), Программы фундаментальных исследований ОНИТ РАН (проект № 2.11), проектов ESTLATRUS 2.1/ELRI-184/2011/14, 1.2/ELRI-121/2011/13.

СПИСОК ЛИТЕРАТУРЫ

1. Методологические вопросы построения имитационных систем: Обзор / *С. В. Емельянов, В. В. Калашников, В. И. Лутков* и др.; Под науч. ред. *Д. М. Гвишиани, С. В. Емельянова*. М.: МЦНТИ, 1973. 87 с.
2. *Охтилев М.Ю., Соколов Б. В., Юсупов Р. М.* Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов. М.: Наука, 2006. 410 с.
3. *Калинин В. Н., Соколов Б. В.* Многомодельный подход к описанию процессов управления космическими средствами // Теория и системы управления. 1995. № 1. С. 56—61.
4. *Соколов Б. В.* Комплексное планирование операций и управление структурами в АСУ активными подвижными объектами. МО: 1992. 232 с.
5. *Цвиркун А. Д., Акинфиев В. К.* Структура многоуровневых и крупномасштабных систем (синтез и планирование развития). М.: Наука, 1993.
6. *Шаппел Д. А.* ESB — Сервисная Шина Предприятия. СПб: БХВ-Петербург, 2008.
7. OASIS Standard: Web Services Business Process Execution Language. 2007.
8. *Laurent S. St., Johnson J., Dumbill E.* Programming Web Services with XML-RPC. CA: O'Reily, 2001.
9. *Vasiliev Y.* SOA and WS-BPEL: Composing Service-Oriented Solution with PHP and Active BPEL. Birmingham, UK: Packt Publishing, 2007.

Сведения об авторе

Семен Алексеевич Потрясаев — канд. техн. наук; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании;
E-mail: spotryasaev@gmail.com

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 519.711.72

Ю. С. Мануйлов, С. В. Зиновьев, Ю. В. Прищепа, Е. Н. Алешин

ИССЛЕДОВАНИЕ ДИНАМИЧЕСКОЙ И ЭНЕРГЕТИЧЕСКОЙ СОВМЕСТИМОСТИ СИСТЕМЫ ПОЗИЦИОНИРОВАНИЯ И УПРАВЛЕНИЯ УГЛОВЫМ ДВИЖЕНИЕМ КОСМИЧЕСКОЙ СОЛНЕЧНОЙ ЭНЕРГОСТАНЦИИ

Исследуется взаимовлияние процессов функционирования системы позиционирования и управления угловым движением космической солнечной энергостанции в рамках решения оптимизационных задач структурно-параметрического синтеза ее основных подсистем.

Ключевые слова: *космическая солнечная энергетическая станция, система позиционирования и управления угловым движением, панель солнечных батарей, концентраторы солнечного излучения.*

Введение. Современные оценки целесообразности создания и использования космических солнечных энергетических станций (КСЭС), предназначенных для передачи энергии на Землю в виде СВЧ-излучения, основаны, главным образом, на результатах анализа и структурно-параметрического синтеза отдельных подсистем: солнечных батарей (СБ), генераторов СВЧ-излучения и активных фазированных антенных решеток (АФАР) [1—3]. При этом оценивание эффективности КСЭС осуществляется, как правило, без учета процесса совместного функционирования данных подсистем, внешних воздействий, а также особенностей энерге-

тического и динамического взаимодействия с другими подсистемами и, в первую очередь, с системой позиционирования и управления угловым движением (СПУУД) элементов конструкции КСЭС. Такой подход в определенной степени справедлив при рассмотрении КСЭС с традиционными планарными солнечными батареями, выходные характеристики которых некритичны к точности ориентации панелей на Солнце. В то же время в качестве альтернативных вариантов рассматриваются СБ с концентраторами солнечного излучения (КСИ). Использование последних позволяет в ряде случаев снизить стоимость и повысить ресурс КСЭС, но при этом требуется существенно более высокая точность их ориентации на Солнце. Необходимость снижения потерь (энергетических, информационных в системах связи, экологических и др.) при передаче СВЧ-излучения наряду с поиском эффективных режимов передачи энергии требует принятия мер, обеспечивающих повышение точности наведения АФАР на приемник. Однако поскольку КСЭС представляет собой многосвязный объект с элементами конструкции ограниченной жесткости, то обеспечение высокоточной ориентации одного из элементов (например, АФАР) усложняет решение этой задачи для других элементов (например, СБ с КСИ). Указанные обстоятельства предопределяют необходимость комплексирования систем на основе решения проблемы энергетической и динамической совместимости СПУУД элементов конструкции КСЭС (АФАР, СБ и др.).

Постановка задачи. При формировании облика КСЭС, характеризуемого вектором параметров $\alpha_i \in \{\alpha_i\}$ ее подсистем, основным требованием является необходимость обеспечения заданного уровня энергоснабжения $P_\tau^\alpha \ge P_{\tau з}^\alpha$, определяемого среднегодовой полезной мощностью P_τ^α на интервале времени τ ($\tau = 1,...,N$, лет) активного существования объекта. Решение данной задачи в оптимизационной постановке предполагает использование показателей экономической эффективности функционирования КСЭС. В качестве одного из таких показателей может быть использована прибыль C, вычисляемая как разность суммарного дохода от реализации энергии C_b^Σ потребителям в течение заданного срока τ и затрат C_l^Σ на обеспечение жизненного цикла функционирования КСЭС:

$$C = C_b^\Sigma(P_\tau^\alpha, c_m^b, \alpha_i) - C_l^\Sigma(P_\tau^\alpha, c_m^l, \alpha_i).$$

Здесь c_m^b — прогнозируемый средний уровень стоимости энергии; c_m^l — прогнозируемый средний уровень стоимости невосполнимых экологических потерь, дополнительных организационно-технических мероприятий, направленных на их предотвращение, а также штрафных санкций, компенсирующих побочные эффекты создания и функционирования КСЭС.

Введем в рассмотрение удельные стоимостные показатели затрат $\overline{c}_i^\varepsilon$, $i = \overline{1,G}$, на обеспечение жизненного цикла G основных подсистем КСЭС. Учитывая затраты, связанные с экологическими и прочими потерями, а также затраты, связанные с мероприятиями по их предотвращению или компенсации, характеризуемые удельными стоимостными показателями $\overline{c}_j^l$, $j = \overline{1,l}$, сформируем комплексный экономический показатель эффективности КСЭС:

$$F(\alpha_i) = \sum_{i=1}^{G} \overline{c}_i^\varepsilon P_\tau^\vartheta(\alpha_i) + \sum_{j=1}^{l} \overline{c}_j^l P_\tau^l(\alpha_i),$$

где $P_\tau^\vartheta, P_\tau^l, P_\tau^l \le P_\tau^\vartheta$, — значения выходной мощности КСЭС и уровня мощности СВЧ-излучения, обусловливающего возникновение отрицательных побочных эффектов функционирования КСЭС.

Поскольку удельные стоимостные показатели $\overline{c}_i^\varepsilon$, $i = \overline{1,G}$, и $\overline{c}_j^l$, $j = \overline{1,l}$, на некотором конечном интервале жизненного цикла КСЭС являются величинами ограниченными:

$\overline{c}_i^{\varepsilon} \le \overline{c}_{i\,з}^{\varepsilon}$, $\overline{c}_j^{l} \le \overline{c}_{j\,з}^{l}$, то оптимизационная задача структурно-параметрического синтеза КСЭС может быть сформулирована как задача математического программирования:

$$\min_{\alpha_i} F(\alpha_i) = \min_{\alpha_i}\left[\sum_{i=1}^{G} \overline{c}_i^{\varepsilon} P_{\tau}^{\vartheta}(\alpha_i) + \sum_{j=1}^{l} \overline{c}_j^{l} P_{\tau}^{l}(\alpha_i)\right];$$

$$P_{\tau}^{\alpha}(\alpha_i) \ge P_{\tau\,з}^{\alpha},\ \alpha_i \in \{\alpha_i\}.$$

Модели совместного функционирования СПУУД элементов конструкции КСЭС. Энергетическая модель функционирования солнечных батарей включает в себя математические модели следующих процессов: концентрации солнечного излучения на поверхности фотопреобразователей (ФП), преобразования его в электрическую энергию, отвода тепла от ФП, деградации характеристик КСИ и ФП под действием высокоэнергетических заряженных частиц радиационных поясов Земли и солнечных космических лучей, а также „собирания“ тока коммутирующими элементами СБ. Так, для системы Кассегрена [1] концентраторов солнечного излучения справедлива аналитическая зависимость коэффициента концентрации излучения k_0 от угла ориентации ν КСИ:

$$k_0 = k_p\left(r_1^2 - r_2^2\right)\left[\exp\left(\gamma^2\nu^2/W^2\right)\right]\Big/ r_2^2,$$

где k_p — коэффициент „перехвата“ излучения при $\nu = 0$; r_1, r_2 — радиусы первичного и вторичного зеркал КСИ; γ, W — коэффициенты аппроксимации.

В расчетной схеме КСИ учитываются линейные и угловые разъюстировки зеркал, неточности ориентации концентраторов на Солнце, статистические неровности отражающих поверхностей, распределение яркости по солнечному диску, а также индикатрисы рассеяния излучения отражающими поверхностями зеркал.

Относительный коэффициент энергосъема [4] для двух установленных на объекте солнечных батарей планарного и концентраторного типов рассчитывается по формуле

$$K = \frac{1}{2}\left(\sum_{i=1}^{2} E_i \,/ E_0\right)\Big/ T,$$

где E_i — усредненный на характерном интервале времени $[0, T]$ энергосъем с учетом неточности ориентации ячеек на Солнце и временной деградации удельных энергетических характеристик СБ указанных типов, E_0 — действительное значение энергосъема для интервала $[0, T]$.

Расчет средней мощности P^{α}, генерируемой ректенной за период T, равный постоянной времени углового движения АФАР, осуществляется путем интегрирования парциальных мощностей элементов ректенны (с учетом схемы их коммутации) в моменты времени фактического приема излучения.

Яркость излучения в заданном направлении φ_B рассчитывается при этом по формуле [5]

$$L(\varphi_B) = \frac{\kappa E_0 \chi_{\Sigma} \delta}{\pi \varphi_{\text{с}}^2 (1 + D)}\left(1 + D\sqrt{1 - \varphi_B / \varphi_{\text{с}}^2}\right),$$

где κ — коэффициент отражения излучения в направлении φ_B, определяемый индикатрисой рассеяния; $\varphi_{\text{с}}$ — угловой размер солнечного диска; χ_{Σ} — функция Хевисайда, характеризующая выполнение совокупности условий видимости солнечного диска из рассматриваемой точки B; δ, D — коэффициенты аппроксимации.

Отдельный энергоизлучающий модуль КСЭС как механическая система представляет собой несущее абсолютно твердое тело и N упругодеформированных элементов (УДЭ), соответствующих системе наименее жестких конструктивных компонентов (СБ и АФАР). С несущим телом связана система координат $Oxyz$, а его движение определяется вектором скорости V_0 поступательного движения полюса O и вектором скорости ω вращения относительно некоторой инерциальной системы координат $O\chi\eta\zeta$.

В соответствии с методом Лурье вектор u смещений элементарных частиц УДЭ объекта определяется как [6]

$$u(\rho) = \sum_{\zeta=1}^{n_0} q_\zeta \Lambda_\zeta(\rho) + \frac{1}{2}\sum_{\zeta=1}^{n_0}\sum_{\xi=1}^{n_0} q_\zeta q_\xi \Lambda_{\zeta\xi}(\rho) + ...,$$

где $\Lambda_\zeta(\cdot)$, $\Lambda_{\zeta\xi}(\cdot)$, $\zeta, \xi = \overline{1, n_0}$, — некоторая система вектор-функций от радиус-векторов ρ положения элементарных частиц УДЭ массой dm, q — обобщенная фазовая координата УДЭ.

Динамика упругого объекта описывается уравнениями Эйлера — Лагранжа. Объемные, поверхностные, линейно-распределенные и/или сосредоточенные силы, действующие на систему УДЭ объекта, рассчитываются в соответствии с заданной схематизацией конструкции.

Результаты исследований. Желаемую динамику корпуса объекта с закрепленным на нем приводным устройством АФАР, к параметрам движения оси диаграммы направленности которой, собственно, и предъявляются высокие требования, зададим опорной траекторией, определенной на множестве решений системы дифференциальных уравнений вида [7]

$$\dot{\varphi}_{\text{оп}} = \Phi(\varphi_{\text{оп}})\omega_{\text{оп}};\ \ \dot{\omega}_{\text{оп}} = U_{\text{г}}(\omega_{\text{оп}}) + U_{\text{оп}},$$

где $\varphi_{\text{оп}}$, $\omega_{\text{оп}}$ — векторы углов и угловых скоростей; $U_{\text{г}}(\omega_{\text{оп}})$, $U_{\text{оп}}$ — векторы обусловленного гироскопическими связями ускорения корпуса и управляющего ускорения соответственно.

Условие квазизатвердевания системы УДЭ [8] запишем в форме

$$\left\| U - U_{\text{оп}} - U_{\text{г}}(\omega_{\text{оп}}) + U_q(\ddot{q}) + U_{\text{вн}} \right\| \le \Theta,$$

где параметр Θ характеризует близость реальной динамики объекта к заданному опорному движению, U — централизованное управление, $U_{\text{вн}}$ — внешние управляющие воздействия.

С позиций теории инвариантного синтеза управление U представляется состоящим из опорного $U_{\text{оп}}$ и „синтезирующего“ $\Delta U = \Delta U(\varphi_{\text{оп}}, \omega_{\text{оп}}, q, \dot{q}, U_{\text{оп}}, U_{\text{г}}, U, U_{\text{вн}})$.

Для снижения уровня возмущений, обусловленных динамикой УДЭ, предлагается за счет „профилирования“ централизованного управления и использования локальных средств активного демпфирования колебаний обеспечивать перевод упругой системы в стационарное состояние, соответствующее действующему нагружению [8]:

$$\ddot{q}_s = 0,\ \dot{q}_s = 0,\ \ s = \overline{1, N}.$$

При этом в идеальном случае может быть обеспечено условие $U_q(\ddot{q}_s,\ s = \overline{1, N}) \equiv 0$.

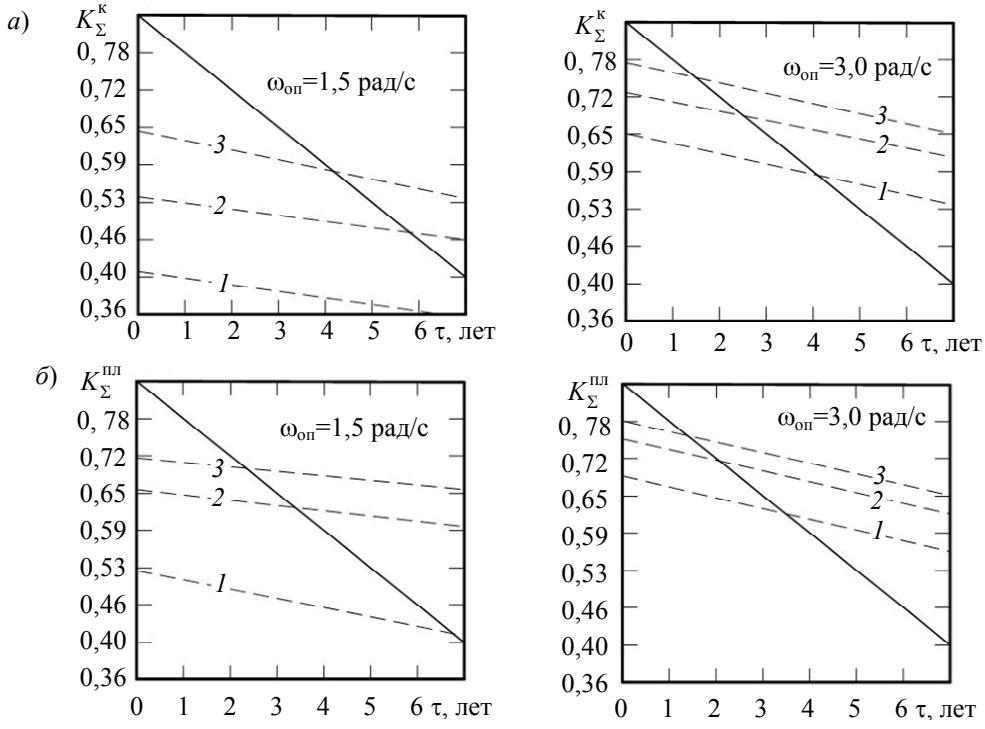
„Синтезирующую“ составляющую предлагается формировать с использованием информации $Y_{\dot{\omega}}$ об угловом ускорении корпуса объекта: $\Delta U = -Y_{\dot{\omega}} + U + U_{\text{г}}(\omega_{\text{оп}})$.

При этом компенсирующая составляющая ΔU в соответствии с принципом двухканальности Петрова формируется по двухконтурной схеме: $\Delta U = \Delta U_{\text{т}} + \Delta U_{\text{гр}}$, где $\Delta U_{\text{т}}$, $\Delta U_{\text{гр}}$ — составляющие, формируемые контурами точного (по отклонению) и грубого (по возмущению) управления. В качестве контура точного управления предлагается использовать пропорционально-дифференциальный регулятор, параметрически оптимизируемый, например, по квадратичному критерию качества.

Оценка эффективности предложенного подхода к формированию облика КСЭС проводилась с использованием разработанного программно-моделирующего комплекса путем

совместного математического моделирования управляемых процессов углового движения корпуса и относительной (в том числе, колебательной) динамики СБ и АФАР КСЭС, а также процессов приема, преобразования и передачи потребителям соответствующих энергетических потоков.

Основная задача экспериментальных исследований заключалась не только в определении структурного и алгоритмического состава СПУУД элементов конструкции КСЭС, при котором обеспечивается преимущество СБ концентраторного типа относительно СБ планарного типа. В ходе экспериментов установлено, что при сопоставимых массогабаритных и удельных энергетических характеристиках обоих типов солнечных батарей возможно обеспечить такую ориентацию фотоприемников на Солнце, при которой СБ концентраторного типа становятся конкурентоспособными уже при сроках активного функционирования объекта от одного года и более. Это объясняется более низким коэффициентом деградации параметров концентраторов по сравнению с планарными панелями. Зависимости коэффициентов относительного энергосъема для СБ концентраторного (*а*) и планарного (*б*) типов $\left(K_{\Sigma}^{\text{к}}, K_{\Sigma}^{\text{пл}}\right)$ от срока активного функционирования объекта при различных значениях $\omega_{\text{оп}}$ представлены на рисунке, где приняты следующие обозначения: *1* — оптимальное управление по фазовому состоянию, *2* — оптимальное управление по расширенному вектору фазового состояния, *3* — управление с учетом „синтезирующей“ составляющей ΔU .



Заключение. Использование разработанного программно-моделирующего комплекса позволяет вплотную подойти к практическому решению задачи оптимального структурно-параметрического синтеза основных подсистем космических солнечных энергостанций при заданных требованиях к выходной мощности, сроку активного существования и допустимому максимально возможному экологическому ущербу от их штатной эксплуатации. Предложенный подход может быть использован для оценки экономической целесообразности реализации и других космических программ, в частности освещения участков земной поверхности в темное время суток.

СПИСОК ЛИТЕРАТУРЫ

1. *Piland R.* The solar power satellite concept evaluation program // Proc. of NASA Conf. “Radiation Energy Conversion in Space”. N. Y., 1978. P. 3—25.

2. *Dockinson R.* SPS microwave subsystem potential impacts end benefits // Proc. of NASA Conf. “Radiation Energy Conversion in Space”. N. Y., 1987. P. 25—35.

3. *Kerwin E. M., Suddath I. H., Arndt G. P.* Antenna optimization and cost consideration for the SPS microwave system // Proc. of IECEC. N. Y., 1981. Vol. 1. P. 272—277.

4. *Armand N. A., Lomakin A. N., Paramonov B. M.* To the solar power satellite accuracy orientation problem // Proc. of Conf. Devoted to Development of K. E. Tciolkovsky's Ideas. M., 1982. P. 123—132.

5. *Monzingo R. A., Miller T. N.* Introduction to adaptive arrays // New York-Chichester-R'islare. Toronto, 1980. 446 p.

6. *Ликинз П.* Уравнения в квазикоординатах для космических аппаратов нежесткой конструкции // Ракетная техника и космонавтика. 1975. Т. 13, № 4. С. 137—140.

7. *Мануйлов Ю. С.* Метод логико-аналитического синтеза в задачах оптимального и адаптивного управления. МО СССР, 1986. 188 с.

8. *Горелов Ю. Н., Мануйлов Ю. С., Шалымов С. В.* Методы реализации принципа квазизатвердевания при стабилизации движения упругих динамических объектов // Методы и алгоритмы исследования и разработки автоматических систем управления: Сб.; Под ред. Л. А. Майбороды. МО СССР, 1989. С. 24—31.

Сведения об авторах

Юрий Сергеевич Мануйлов — д-р техн. наук, профессор; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург;
E-mail: ymanoff@yahoo.com, kotmanoff@rambler.ru

Сергей Валерьевич Зиновьев — канд. техн. наук, доцент; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург; E-mail: zinoviev_sv@mail.ru

Юрий Владимирович Прищепа — канд. техн. наук; Филиал открытого акционерного общества «Концерн радиостроения „Вега“», Санкт-Петербург; директор;
E-mail: mail@spb.vega.su

Евгений Николаевич Алешин — канд. техн. наук; Военно-космическая академия им. А. Ф. Можайского, кафедра автоматизированных систем управления космическими комплексами, Санкт-Петербург; E-mail: aleshin_evgeny@inbox.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 004.021

А. М. Тележкин

СИСТЕМА САМПО+ ДЛЯ СОЗДАНИЯ И АНАЛИЗА ИСТОРИЧЕСКОЙ БАЗЫ ДАННЫХ

Представлена методология создания исторических баз данных, предназначенных для более точной оценки необходимых ресурсов для инициируемых проектов. Приведена процедура создания такой базы с помощью системы САМПО+.

Ключевые слова: *историческая база данных проекта, сбор и анализ проектных метрик.*

Введение. Компании, работающие в сфере программного обеспечения, достигают определенного уровня зрелости, когда накапливается некоторый объем информации об уже выполненных в компаниях проектах [1]. Причем эти данные не всегда используются для оценки ресурсов при запуске следующих проектов. Сбор и накопление информации по реализованным проектам в исторической базе данных (ИБД) обеспечивает возможность использования имеющегося опыта для более точного планирования и контроля новых проектов.

Историческая база данных о выполненных проектах содержит такие характеристики, как: предметная область проекта, модель жизненного цикла, используемые технологии, затраченные ресурсы, риски, бюджет, цели проекта, даты начала и окончания этапов проекта и т.д., а также различные процессные и продуктовые метрики, связанные с исполнением проектов в конкретной организации. Использование ИБД позволяет на основе анализа найденных проектов-аналогов произвести оценку ресурсов, необходимых для успешного завершения инициируемого или выполняемого нового проекта.

Проект-аналог — это проект, который, по мнению эксперта, может служить в качестве основы для оценивания соответствующих характеристик исследуемого проекта, а в предельном случае и заимствования модели процесса разработки программного изделия.

Система САМПО+. Система поддержки создания ИБД компаний (САМПО+), разрабатывающих программное обеспечение (далее — Система), предназначена для снижения трудозатрат и оценки необходимых ресурсов. Система позволяет на основе ИБД сформировать модель базы данных для конкретной организации, а также произвести объективную оценку выполнимости инициируемых проектов. В этом качестве Система служит инструментом для снижения риска незавершения нового проекта.

Наряду со своим прямым назначением Система может использоваться в качестве средства повышения уровня профессиональных знаний сотрудников компании, а также в учебном процессе при подготовке специалистов по разработке программного обеспечения.

Пользователем Системы является эксперт в области управления процессом разработки программных изделий, владеющий знаниями, плохо поддающимися формализации. В Системе используется методология моделирования и принятия решений (прозрачные технологии) на основе алгоритмических сетей, разработанная в СПИИРАН [2].

Базу прозрачной технологии в системе САМПО+ составляет алгоритм вычисления оценок, предложенный в начале 1970-х гг. акад. Ю. И. Журавлёвым [3]. Основная идея этого алгоритма заключается в том, что для определения класса какого-либо объекта используются не отдельные характеристики, а их совокупность (ансамбль).

Важной особенностью Системы является хранение ссылок на документы, из которых взята информация, что связано с необходимостью обеспечения возможности контроля и повторного анализа вводимых значений характеристик.

Методология, на основе которой в Системе определяется экспертное множество характеристик, формирующих ИБД, была предложена и испытана в СПИИРАН и включает следующие аналитические процессы:

вербальный — формирование, исходя из экспертного множества, исходных множеств источников, характеристик и проектов на основе анализа методической и специальной литературы, а также бизнес-процессов компании;

количественный — формирование уточненных множеств источников, характеристик и проектов на основе анализа обеспеченности каждого из них проектной информацией и представления об их значимости для поиска проектов-аналогов;

качественный — формирование рекомендуемых множеств источников, характеристик и проектов на основе экспериментов, подтверждающих достаточность сформированного в ходе предыдущих процессов множества характеристик, для решения задачи распознавания (под задачей распознавания в данном случае понимается поиск проекта-аналога).

Подробно проблемы формирования ИБД, а также методология их построения на примере системы САМПО+ рассмотрены в работе [4]. В настоящей статье рассматриваются уточнения к методологии, а также особенности формирования и исследования ИБД в системе САМПО+ по опыту ее практического применения в организации “Exigen Services” (Санкт-Петербург) в ходе анализа базы выполненных проектов.

Режимы Системы. Система обеспечивает поддержку режимов формирования ИБД, исследования ИБД и режима использования собранной информации в ИБД.

Главное меню Системы представлено схемой, приведенной на рис. 1.

Рис. 1

Множество источников. Результатом анализа информации для создания ИБД является множество источников, в качестве которых могут быть названы знания эксперта, метрическая/корпоративная база данных компании, а также документы, создаваемые на основе шаблонов, и сами шаблоны.

Атрибуты источника: наименование; описание; множество, к которому принадлежит источник (исходное, уточненное, рекомендуемое); используемость во множестве характеристик; используемость во множестве проектов; дата и автор последнего изменения.

Множество характеристик. Данное множество позволяет описать любой проект, реализованный или выполняемый в компании, для последующего анализа и использования фактологических данных при инициации новых проектов.

Для формирования исходного множества характеристик используются, как правило, пять источников:

— общетехническая литература, содержащая информацию о выполнении проектов любых типов;

— специальная литература, содержащая информацию о выполнении проектов разработки программных изделий (CMM, CMMI, ISO и пр.);

— база данных компании, в которой собраны какие-либо процессные, продуктовые и проектные метрики;

— документы по проекту, или имущество проекта, которые формируются во время работы над проектом, а также по его завершении;

— мнения экспертов.

Для уменьшения пространства поиска проектов-аналогов предлагается структурировать исходное множество характеристик, а именно, выделить в нем три подмножества: „категории“, „интегральные характеристики“ и „терминальные характеристики“.

Атрибуты характеристики: уникальный номер; наименование; описание; тип (интегральная, вычисляемая, число, текст, шкала, дата); описание области задания; значение по умолчанию; документы, из которых характеристика может быть извлечена (заключение-мнение эксперта, корпоративная база данных, документы проекта); уникальный номер родителя; уникальные номера потомков; формула, по которой вычисляется значение характеристики; множество, к которому принадлежит характеристика (исходное, уточненное, рекомендуемое); используемость в проектах; дата и автор последнего изменения.

Множество проектов. Данное множество содержит все проекты организации, которые были выполнены, а также выполняются в настоящее время.

Атрибуты проекта: наименование; содержание; перечень источников информации; местонахождение источников; значения характеристик; множество, к которому принадлежит проект (исходное, уточненное, рекомендуемое); обеспеченность информацией; дата и автор последнего изменения.

Организация работы Системы. Добавление, редактирование и удаление источников, характеристик и проектов исходного, уточненного и рекомендуемого множеств организовано по единому сценарию. При входе в соответствующий режим Системы в выпадающем списке предлагается выбрать одно из множеств, которое будет редактироваться (исходное, уточненное или рекомендуемое). В зависимости от выбора в таблицу данных загружается необходимое множество.

Ввод значений, характеризующих конкретное множество, осуществляется по другой схеме. Пользователю предлагается таблица, в которой указываются наименования проектов, характеристик и их значения. При этом возможны следующие действия пользователя.

1. В графе наименований характеристик выбирается одна из характеристик, после чего открывается форма ввода значений, и пользователь указывает значения этой характеристики по всем проектам.

2. В графе наименований проектов выбирается один из проектов, после чего открывается форма ввода значений, и пользователь последовательно указывает значения характеристик для данного проекта.

3. Выбирается произвольная ячейка, содержащая значения характеристик по конкретному проекту, после чего открывается форма ввода значений, и пользователь указывает значение конкретной характеристики для конкретного проекта.

Режимы исследования. В Системе заданы четыре режима исследования накопленной информации — источников, характеристик, проектов и функциональной пригодности.

В режиме *исследования источников* пользователь работает с таблицей, в которой содержится информация о том, сколько раз данный источник был использован в том или ином проекте, а также процентное и абсолютное число „вхождений“ каждого из источников во все проекты.

В режиме *исследования характеристик* пользователь работает с таблицей, в которой содержится информация о значениях характеристик по каждому проекту, а также процентное и абсолютное число „вхождений“ каждой из характеристик во все проекты.

В режиме *исследования проектов* пользователь работает с таблицей, в которой содержится информация о том, сколько источников было использовано в каждом проекте, а также процентное и абсолютное число „вхождений“ источников в каждый проект.

В режиме *исследования функциональной пригодности* базы данных пользователь работает с таблицей множества характеристик, с помощью которой он может построить свое подмножество характеристик и определить проекты, аналогичные инициируемому. Работа в данном режиме продемонстрирована на рис. 2 в виде снимка с экрана компьютера.

Рис. 2

Заключение. В результате исследований было сформировано множество из 360 различных характеристик, которые достаточно полно описывают процесс разработки программного обеспечения для стандартной компании.

Система САМПО+ использовалась при анализе характеристик для создания БД, собранных компанией “Exigen Services” в ходе выполнения проектов и при их завершении. Исследование проводилось в 2009—2010 гг., в общей сложности было рассмотрено 342 проекта, из них в Систему включен 71 проект.

Система также использовалась при анализе базы данных автоматизации документооборота НП „Объединение подземных строителей“ (Санкт-Петербург).

Система САМПО+ создана в среде MS Excel 2007/2010 и содержит 6200 строк программного кода на языке Visual Basic for Applications.

Статья подготовлена по результатам работы, выполненной при финансовой поддержке ведущих университетов Российской Федерации (субсидия 074-U01 — Университет ИТМО).

СПИСОК ЛИТЕРАТУРЫ

1. *Баранов С. Н., Домарацкий А. Н., Ласточкин Н. К., Морозов В. П.* Процесс разработки программных изделий. М.: Наука — Физматлит, 2000. 176 с.
2. *Морозов В. П.* Поддержка принятия решений, ориентированная на знания эксперта // Тр. XII Санкт-Петербург. междунар. конф. „Региональная информатика (РИ-2010)“, 20—22 окт. 2010 г. СПб: СПОИСУ, 2011. С. 69—73.
3. *Журавлев Ю. И., Никифоров В. В.* Алгоритмы распознавания, основанные на вычислении оценок // Кибернетика. 1971. № 3. С. 1—11.
4. *Тележкин А. М.* Создание исторических баз данных при помощи системы САМПО+ // Тр. Юбилейной XIII Санкт-Петербург. междунар. конф. „Региональная информатика (РИ-2012)“, 24—26 окт. 2012 г. СПб: СПОИСУ, 2013. С. 84—90.

Сведения об авторе

Александр Михайлович Тележкин — аспирант; СПИИРАН, лаборатория информационных технологий в системном анализе и моделировании; E-mail: telezhkin@gmail.com

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 004.056

А. В. ФЕДОРЧЕНКО, А. А. ЧЕЧУЛИН, И. В. КОТЕНКО

ПОСТРОЕНИЕ ИНТЕГРИРОВАННОЙ БАЗЫ УЯЗВИМОСТЕЙ

Представлены результаты исследования открытых баз данных уязвимостей и описание процесса их интеграции для применения в системах оценивания защищенности компьютерных сетей. Предлагаются модель процесса формирования и структура интегрированной базы уязвимостей, а также описание и анализ разработанного прототипа.

Ключевые слова: *анализ защищенности, базы данных уязвимостей, системы мониторинга безопасности.*

Введение. В настоящее время существует большое количество баз данных (БД) уязвимостей, как открытых для общего доступа, так и закрытых, используемых в коммерческих продуктах. Они применяются в различных системах безопасности, сканерах уязвимостей и других средствах обеспечения комплексной защиты компьютерных систем. Однако применение таких баз данных в системах оценивания защищенности компьютерных сетей в режиме реального времени недопустимо вследствие низкой скорости поиска записей уязвимостей для оперативной обработки событий, нарушающих информационную безопасность [1—4]. Также следует отметить, что формирование БД уязвимостей не стандартизовано и производится несогласованно, что влияет на точность обнаружения уязвимостей в используемом программно-аппаратном обеспечении.

Для увеличения объема уникальных записей уязвимостей и списков программно-аппаратных продуктов, соответствующих этим уязвимостям, предлагается объединение открытых баз уязвимостей. Реализация данного процесса предусматривает разработку методики интеграции баз уязвимостей и проектирование структуры интегрированной базы для адаптации ее к быстрому поиску записей уязвимостей. Конечное использование формируемой интегрированной базы уязвимостей подразумевает ее эксплуатацию в системах оценивания защищенности компьютерных сетей, анализ которых должен проводиться в режиме, близком к

реальному времени. При успешной реализации задачи интеграции баз уязвимостей и, как следствие, задачи формирования интегрированной базы предполагается повышение эффективности работы систем оценивания защищенности.

Анализ открытых баз уязвимостей. Для формирования интегрированной базы уязвимостей был проведен анализ ряда открытых баз уязвимостей, таких как: Общие уязвимости и воздействия (Common Vulnerabilities and Exposures — CVE) [5], Национальная база данных уязвимостей (National Vulnerabilities Database — NVD) [6], Открытая база данных уязвимостей (Open Source Vulnerabilities Data Base — OSVDB) [7].

Анализ базы данных CVE показал, что она обладает высокой степенью связности с другими источниками описания уязвимостей. Однако недостатком данной базы является отсутствие в описании уязвимостей спецификации программно-аппаратного обеспечения, для которого характерна эта уязвимость.

Более подробное описание содержащихся в базе CVE уязвимостей представлено в базе данных NVD. Эта база включает в себя описание уязвимостей с указанием подверженных им программно-аппаратных продуктов в формате CPE. Кроме того, данная база содержит показатели, характеризующие уязвимости в формате Общей системы оценивания уязвимостей (Common Vulnerability Scoring System — CVSS) [8]. Однако количество ссылок на другие базы данных, по которым производится интеграция записей уязвимостей, в базе NVD меньше, чем в базе CVE (последняя включает все ссылки базы NVD).

Следует также отметить, что база NVD единственная из исследуемых баз, содержащая логические описания конфигураций программно-аппаратного обеспечения для каждой уязвимости. Данные конфигурации определяют уязвимые системы и представляют собой списки программно-аппаратного обеспечения, объединенные логическими операторами И/ИЛИ. Считается, что конфигурация содержит зависимость в том случае, если она содержит логический оператор И (т.е. определяет уязвимую конфигурацию, содержащую одновременно не менее двух продуктов). Анализ базы NVD показал, что из всех уязвимостей зависимости содержат только 10,63 % записей. Причем каждая из этих зависимостей содержит только один логический оператор И. Такой невысокий процент зависимостей обусловливает возможность хранения конфигураций программно-аппаратного обеспечения в одной таблице, что позволяет значительно снизить вычислительные ресурсы, необходимые для поиска подходящих уязвимостей.

При анализе базы данных OSVDB было выявлено множество ссылок на сторонние источники, причем наилучшими показателями по количеству и уникальности обладают ссылки на базы CVE и NVD.

По результатам проведенного анализа именно рассмотренные базы данных стали основными источниками информации об уязвимостях для формирования интегрированной базы.

Модель процесса интеграции данных открытых баз уязвимостей. Данные, хранящиеся в рассмотренных открытых базах уязвимостей и необходимые для системы оценивания защищенности, условно разделяются на две группы: 1) описания уязвимостей (набор характеристик, указывающих на причины их возникновения, условия успешной реализации уязвимости и способы ее устранения); 2) описания конфигурации уязвимых продуктов, для которых данная уязвимость характерна. На этой основе в процессе интеграции открытых баз уязвимостей выделяются две составляющие: интеграция записей уязвимостей и интеграция описаний продуктов.

Интеграция записей уязвимостей производится по указанным в описании ссылкам, которые делятся на две категории: 1) прямые, т.е. непосредственно указывающие на источник информации (идентификатор записи уязвимости), используемый при интеграции; 2) косвенные, указывающие на источники информации, не используемые при интеграции. Для соответствия уязвимостей, выявленных по прямым ссылкам, необходимо и достаточно наличие одного указания на используемые источники, а для соответствия по косвенным ссылкам —

наличие двух указаний. Таким образом, формирование интегрированного списка уязвимостей можно представить в виде модели, приведенной на рис. 1.

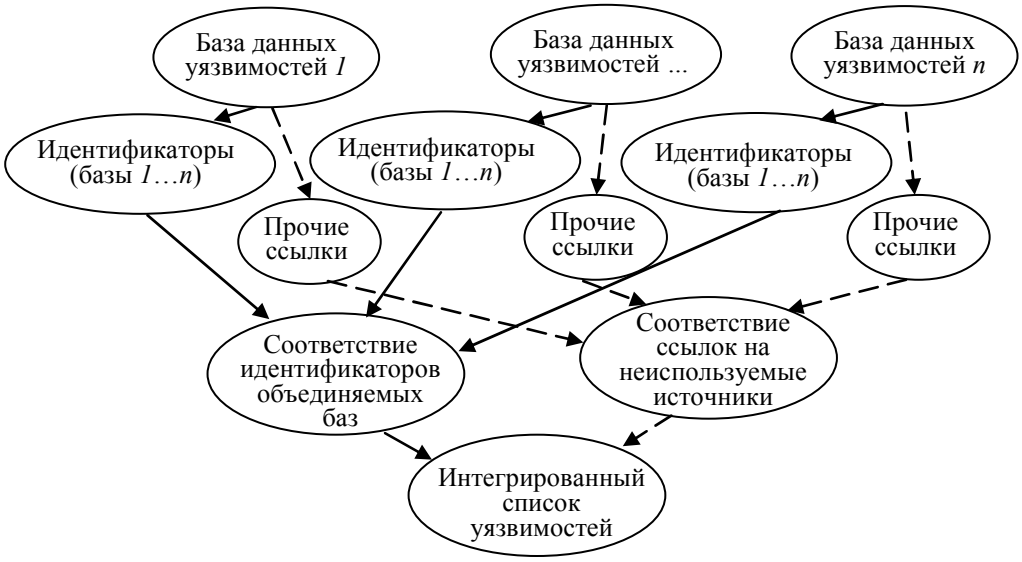


Рис. 1

Следует отметить, что если уязвимости в пределах одной базы содержат ссылки на записи из этой же базы, то такие уязвимости объединяются в одну запись интегрированного списка, что обеспечивает уникальность каждой записи уязвимости в формируемой базе.

При формировании интегрированного словаря продуктов за основу берется схема „Общее перечисление платформ“ (Common Platform Enumeration — CPE) [9]. CPE — это структурированная схема именования компьютерных систем и платформ, основанная на синтаксисе Универсальных идентификаторов ресурсов (Uniform Resource Identifiers — URI) [10]. Выбор словаря CPE обусловлен тем, что он содержит большее количество описаний продуктов по сравнению с аналогичными словарями, а формат представления продуктов является лучшим на данный момент. На первом этапе интеграции описаний продуктов производится сравнение записей, содержащихся в словаре CPE, с записями, имеющимися в пространствах имен используемых баз уязвимостей. Затем оба вида записей объединяются в список и в результате формируют интегрированный словарь продуктов. Модель процесса формирования интегрированного словаря продуктов представлена на рис. 2.

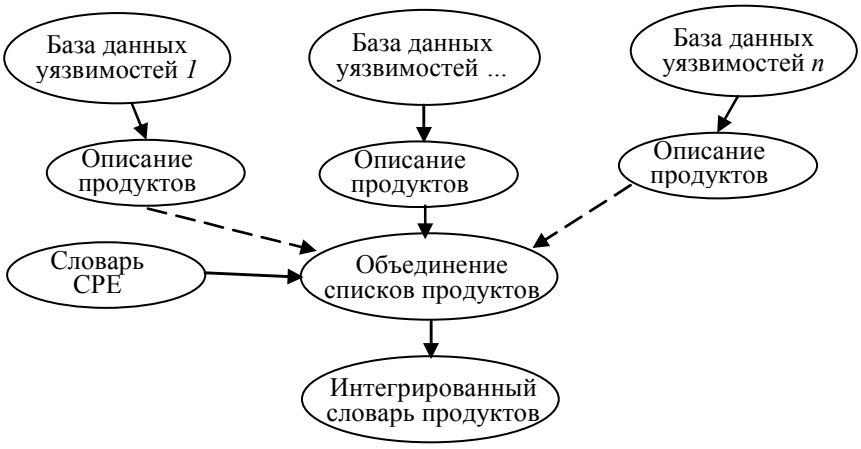


Рис. 2

Несмотря на то, что в качестве основы для формирования интегрированного словаря продуктов был использован словарь CPE, формат самих записей был изменен и приобрел следующий вид:

{тип}:{производитель}:{продукт}:{версия}:{модификация}:{ редакция}.

Структура интегрированной базы уязвимостей. На основе предложенной модели интеграции записей уязвимостей и списков продуктов была разработана структура интегрированной базы уязвимостей. Под структурой в данном случае понимается реляционная модель базы данных [11]. В результате выполнения процесса интеграции вся информация об уязвимостях и программно-аппаратном обеспечении записывается в реляционные таблицы 1) производителей, 2) продуктов, 3) конечных описаний продуктов (включает поля „тип“, „версия“, „модификация“, „редакция“) и 4) уязвимостей, а также таблицу связности (5) записей уязвимостей и описаний продуктов (включает поле „параметр зависимости“): см. рис 3. При этом связь содержащихся в таблицах данных осуществляется за счет использования первичных ключей PK (однозначно определяющих каждую запись в таблице) и внешних ключей K (хранящих значение первичного ключа из другой таблицы).

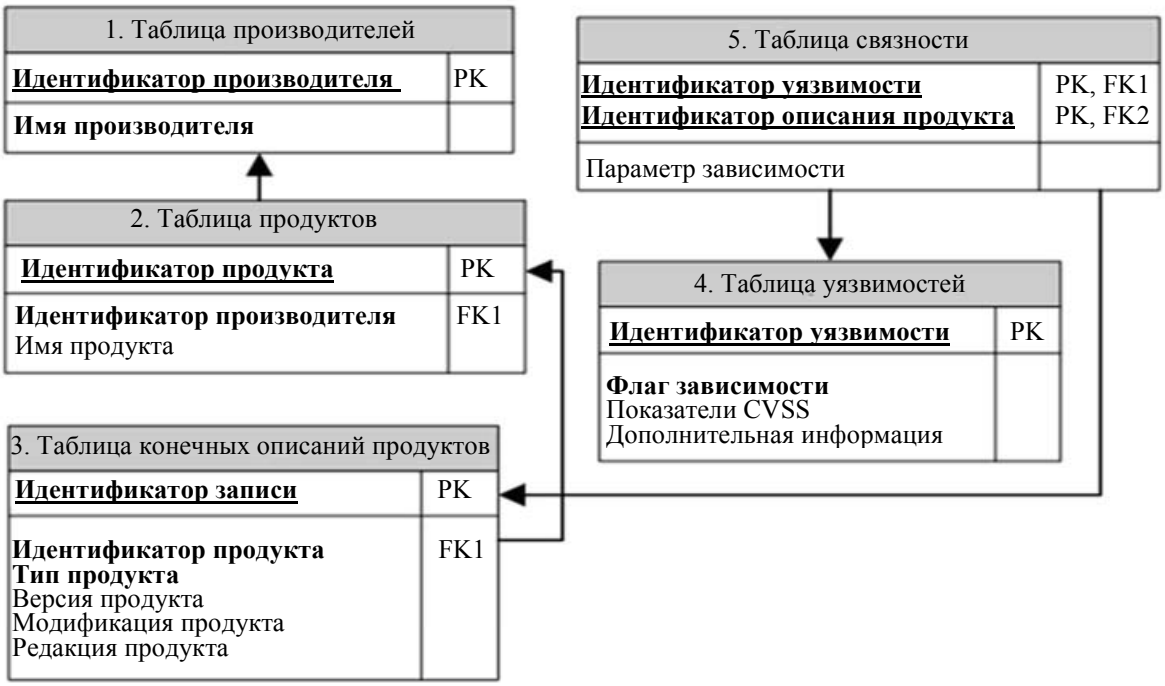


Рис. 3

Параметр зависимости в 5-й реляционной таблице (см. рис. 3) является единственным признаком, определяющим связь между уязвимым программно-аппаратным обеспечением и продуктами, при которой уязвимость может быть реализована. Таким образом, если указанный в качестве ключа продукт имеет уязвимость без такой зависимости, то поле „параметр зависимости“ этой записи будет иметь значение „0“. Если же уязвимость, характерная для некоторого продукта, реализуется только в случае наличия другого продукта, то такая зависимость описывается следующим образом:

— группа продуктов, для которых могут быть реализованы уязвимости в зависимости от наличия в системе любого из продуктов другой группы, является группой *A*;

— группа продуктов, влияющих на возможность реализации уязвимости продуктов группы *A*, является группой *B*;

— значение поля „параметр зависимости“ (в 5-й реляционной таблице) для продуктов группы *A* будет нечетным, начиная со значения „1“;

— значение поля „параметр зависимости“ (в 5-й таблице) для продуктов группы *B* будет четным, начиная со значения „2“.

Рассмотренная структура интегрированной базы данных уязвимостей позволяет с помощью единственного SQL-запроса к ней осуществлять поиск уязвимостей, присущих программно-аппаратным платформам, задаваемым в качестве списка идентификаторов описаний продуктов при реализации запроса.

Прототип интегрированной базы уязвимостей. В качестве практической реализации интегрированной БД уязвимостей был разработан прототип, для формирования которого использовались следующие источники описания уязвимостей и продуктов: CVE, CPE, NVD, OSVDB. Для оценки эффективности построения интегрированной БД уязвимостей следует воспользоваться количественными показателями, приведенным в табл. 1.

Таблица 1

Источник данных	Общее количество записей уязвимостей	Общее количество описаний продуктов	Количество уникальных записей уязвимостей
CVE	67 966	—	48 621
CPE	—	82 987	—
NVD	59 779	147 505	48 621
OSVDB	98 625	127 564	59 827
Интегрированная база уязвимостей	**73 908**	**183 321**	**73 908**

Опираясь на представленные результаты, можно вывести показатель (P), характеризующий преимущество интегрированной базы данных относительно использованных для ее формирования источников информации. Результаты сравнительного анализа представлены в табл. 2.

Таблица 2

Источник данных	P, %	
	Интегрированный список уязвимостей	Интегрированный словарь продуктов
CVE	34	—
CPE	—	55
NVD	34	20
OSVDB	19	30

Представленный прототип интегрированной базы данных был использован как базовый компонент для построения автоматизированной системы моделирования атак и анализа защищенности компьютерных сетей [4, 12]. Для проверки эффективности функционирования интегрированной БД была проведена серия экспериментов по моделированию различных сценариев атак.

Заключение. Результаты анализа открытых баз уязвимостей продемонстрировали имеющиеся в них проблемы, которые, в свою очередь, негативно сказываются на процессе мониторинга компьютерных систем на предмет наличия уязвимого программного и аппаратного обеспечения. Для преодоления выявленных проблем в данной статье предложена модель интеграции данных открытых баз уязвимостей, определены форматы данных и осуществлено проектирование структуры интегрированной базы данных уязвимостей. В качестве практической реализации разработан ее прототип.

Эксперименты, проведенные с использованием прототипа, доказали преимущество и конкурентоспособность сформированной интегрированной базы по отношению к лидерам в области баз уязвимостей, что делает ее подходящей в качестве компонента для использования в системах оценивания защищенности.

Направлением дальнейших исследований является разработка и модификация интегрированной базы данных уязвимостей за счет увеличения числа используемых открытых баз, присвоения показателей доверия источникам ссылок и расширенного анализа уникальности уязвимостей для обеспечения наиболее качественной интеграции.

Статья подготовлена по результатам работы, выполняемой при финансовой поддержке Российского фонда фундаментальных исследований (гранты 13-01-00843, 13-07-13159, 14-07-00697, 14-07-00417, 14-37-50735), программы фундаментальных исследований ОНИТ РАН (контракт № 2.2), проекта ENGENSEC программы Европейского сообщества TEMPUS и государственных контрактов № 14.604.21.0033, 14.604.21.0137, 14.604.21.0147 и 14.616.21.0028.

СПИСОК ЛИТЕРАТУРЫ

1. *Kotenko I., Stepashkin M.* Network security evaluation based on simulation of malefactor's behavior // Proc. of the Intern. Conf. on Security and Cryptography (SECRYPT'06). Setubal, Portugal, 2006. P. 339—344.
2. *Котенко И. В., Степашкин М. В., Богданов В. С.* Архитектуры и модели компонентов активного анализа защищенности на основе имитации действий злоумышленников // Проблемы информационной безопасности. Компьютерные системы. 2006. № 2. С. 7—24.
3. *Ruiz J. F., Harjani R., Maña A., Desnitsky V., Kotenko I., Chechulin A.* A methodology for the analysis and modeling of security threats and attacks for systems of embedded components // Proc. of the 20th Euromicro Intern. Conf. on Parallel, Distributed and Network-Based Processing (PDP 2012). Garching, Germany. 2012. C. 261—268.
4. *Kotenko I. V., Chechulin A. A.* Common framework for attack modeling and security evaluation in SIEM systems // IEEE Intern. Conf. on Green Computing and Communications, Conf. on Internet of Things, and Conf. on Cyber, Physical and Social Computing; Besançon, France, 11—14 Sept., 2012; Los Alamitos, CA: IEEE Computer Society, 2012. P. 94—101.
5. Common Vulnerabilities and Exposures (CVE) [Электронный ресурс]: <http://cve.mitre.org/>, 05.05.2014.
6. National Vulnerabilities Database (NVD) [Электронный ресурс]: <http://nvd.nist.gov/>, 05.05.2014.
7. Open Source Vulnerabilities Data Base (OSVDB) [Электронный ресурс]: <http://osvdb.org/>, 05.05.2014.
8. Common Vulnerability Scoring System (CVSS) [Электронный ресурс]: <http://www.first.org/cvss>, 05.05.2014.
9. Common Platform Enumeration (CPE) [Электронный ресурс]: <http://cpe.mitre.org>, 05.05.2014.
10. *Котенко И. В., Дойникова Е. В.* Анализ протокола автоматизации управления данными безопасности SCAP // Защита информации. Инсайд. 2012. № 2. С. 56—63.
11. *Фиайли К.* SQL “Quick Start”. М.: ДМК Пресс, 2003. 456 с.
12. *Kotenko I., Chechulin A.* Attack modeling and security evaluation in SIEM systems // Intern. Transact. on Systems Science and Applications. 2012. Vol. 8, Dec. P. 129—147.

Сведения об авторах

Андрей Владимирович Федорченко — СПИИРАН, лаборатория проблем компьютерной безопасности; мл. научный сотрудник; E-mail: fedorchenko@comsec.spb.ru

Андрей Алексеевич Чечулин — канд. техн. наук; СПИИРАН, лаборатория проблем компьютерной безопасности; E-mail: chechulin@comsec.spb.ru

Игорь Витальевич Котенко — д-р техн. наук, профессор; СПИИРАН, лаборатория проблем компьютерной безопасности; E-mail: ivkote@comsec.spb.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 681.3

А. В. Муравьев, А. Н. Березин, Д. Н. Молдовян

ПРОТОКОЛ СТОЙКОГО ШИФРОВАНИЯ СООБЩЕНИЙ С ИСПОЛЬЗОВАНИЕМ КОРОТКИХ КЛЮЧЕЙ

Предложены способ и протокол гарантированной защиты информации, передаваемой по открытым каналам, путем ее криптографического преобразования при использовании разделяемых секретных ключей малого размера (до 56 бит).

Ключевые слова: *шифрование, криптографические протоколы, секретный ключ, стойкость, задача дискретного логарифмирования, коммутативный шифр.*

Введение. Применяемые в системах защиты информации криптографические схемы с секретным ключом (одноключевые) обеспечивают гарантированную стойкость шифрования сообщений при использовании ключей достаточно большого размера, например 128 или 256 бит. На практике же существует необходимость срочной передачи конфиденциальной информации, когда и отправитель, и получатель имеют ключи лишь малого размера (от 32 до 56 бит). Использование таких ключей непосредственно в алгоритмах симметричного шифрования позволяет потенциальному нарушителю определить эти ключи методом полного перебора по ключевому пространству. В этом случае возникает необходимость обеспечения приемлемого уровня стойкости шифрования, например, равного 2^{128} операциям.

Для решения данной задачи следует включить в процесс шифрования алгоритм коммутативного шифрования, не требующий использования разделяемых (общих) секретных ключей [1, 2] (так называемое бесключевое шифрование, которое позволяет обеспечить необходимый уровень стойкости). Недостатком данного алгоритма является невозможность обеспечить аутентификацию сообщений. В предлагаемых протоколах для аутентификации сообщений используется разделяемый секретный ключ малого размера, благодаря чему потенциальный нарушитель не может выдать себя за отправителя или получателя сообщений, также он не имеет вычислительной возможности определить секретный ключ методом полного перебора.

Протокол передачи сообщения без обмена ключами. В качестве процедуры коммутативного шифрования возможно использование трехпроходного протокола Шамира [3], что позволяет передать секретное сообщение по открытому каналу связи без использования процедуры распределения ключей. В основе протоколов данного типа лежит стойкий алгоритм коммутативного шифрования, для которого выполняется условие

$$E_A\left(E_B\left(M\right)\right)=E_B\left(E_A\left(M\right)\right),$$

где E — функция криптографического преобразования, A и B — неразделяемые секретные ключи отправителя и получателя соответственно, M — преобразуемое сообщение.

При использовании данного протокола пересылка сообщения M по открытому каналу связи осуществляется следующим образом.

1. Отправитель шифрует сообщение M по своему ключу A и посылает его получателю: $C_1=E_A(M)$.

2. Получатель шифрует криптограмму C_1 по своему ключу B и посылает отправителю: $C_2=E_B(C_1)=E_B(E_A(M))$.

3. Отправитель, используя процедуру расшифровывания D по своему ключу A, преобразует криптограмму C_2 и посылает получателю: $C_3=D_A(C_2)=D_A(E_B(E_A(M)))=D_A(E_A(E_B(M)))=$ $=E_B(M)$. Исходя из полученного шифр-текста C_3, получатель, используя процедуру рас-

шифровывания D по своему ключу B, восстанавливает сообщение M по формуле $M = D_B(E_B(M))$.

Ключи A и B могут выбираться произвольно, и для каждого передаваемого сообщения возможна выработка новых пар ключей. Очевидно, что обмен ключами не происходит, поэтому данный протокол может называться протоколом бесключевого шифрования. Однако протоколы данного типа уязвимы для атаки „человек посередине“, т.е. потенциальный нарушитель может выдавать себя как легальный участник протокола.

Новый способ применения бесключевого шифрования. Шифрование сообщений по разделяемому ключу малого размера не является безопасным, так как при перехвате криптограммы практически возможно нахождение ключа путем перебора по ключевому пространству. Для обеспечения требуемого уровня стойкости представляется уместным передача сообщения с использованием протокола бесключевого шифрования и аутентификация сообщений, выполняемая по разделяемому ключу малого размера. Причем такой вариант применения разделяемого секретного ключа принципиально отличается от его применения в процедурах шифрования сообщений, так как у потенциального нарушителя будет лишь одна попытка угадать секретный ключ и навязать ложное сообщение, тогда как при шифровании у него имеется возможность многократного опробования различных значений ключа. Вероятность обмана в случае применения секретного ключа для аутентификации составляет 2^{-k}, где k — длина ключа (в битах). Например, при использовании 32-битового ключа вероятность составит 2^{-32}, что пренебрежимо мало даже для достаточно критичных применений. Благодаря этому представляется возможным использовать ключи малого размера для аутентификации сообщений в протоколах бесключевого шифрования. При практической реализации необходимо обеспечить неразрывность процедуры аутентификации и бесключевого шифрования.

Аутентификация шифр-текстов. Значения шифр-текстов C_1, C_2, C_3, получаемых в результате выполнения протокола коммутативного шифрования, являются вычислительно неотличимыми от случайных значений. Шифрование криптограмм по разделяемому короткому ключу с использованием симметричного алгоритма $G_K(C)$, где G_K — алгоритм симметричного шифрования [4] по ключу K, не позволяет потенциальному нарушителю найти значение разделяемого короткого ключа, а для легального получателя появляется возможность аутентификации отправителя шифр-текстов. При использовании такого подхода протокол стойкого шифрования по ключу малого размера выглядит следующим образом.

1. Отправитель шифрует сообщение M по своему неразделяемому ключу A: $C_1 = E_A(M)$; полученную криптограмму C_1 зашифровывает по разделяемому секретному ключу K с использованием алгоритма симметричного шифрования [4]: $S_1 = G_K(C_1)$; полученное значение S_1 отправляет получателю.

2. Получатель расшифровывает шифр-текст S_1 по разделяемому ключу K и получает значение C_1: $C_1 = G^{-1}{}_K(S_1)$; шифрует криптограмму C_1 по своему неразделяемому ключу B: $C_2 = E_B(C_1)$; полученную криптограмму C_2 зашифровывает по разделяемому секретному ключу K с использованием алгоритма симметричного шифрования [4]: $S_2 = G_K(C_2)$; полученное значение S_2 посылает отправителю.

3. Отправитель расшифровывает шифр-текст S_2 по разделяемому ключу K и получает значение C_2: $C_2 = G^{-1}{}_K(S_2)$; затем, используя процедуру расшифровывания D по своему неразделяемому ключу A, преобразует криптограмму C_2 и посылает получателю: $C_3 = D_A(C_2) = E_B(M)$.

Получатель расшифровывает сообщение M из шифр-текста C_3: $M = D_B(E_B(M))$. Использование на первых двух шагах протокола дополнительного симметричного шифрования по общему разделяемому ключу позволяет предотвратить атаки со стороны активного нарушителя, т.е. происходит взаимная аутентификация пересылаемого сообщения получателем и отправителем.

В качестве функции криптографического преобразования $E_K(M)$, обеспечивающей свойство коммутативности, может использоваться алгоритм шифрования Полига — Хеллмана [2], основанный на вычислительной трудности задачи дискретного логарифмирования по модулю простого числа. Базовой операцией в данном протоколе является операция возведения в степень по модулю большого простого числа p. Шифрование сообщения $M < p$ выполняется путем возведения его в некоторую степень e, взаимно простую с числом $p-1$: $C = E(M) = M^e \bmod p$. Криптограмма C расшифровывается посредством возведения ее в степень d: $M = D(C) = C^d \bmod p$. Выбор степени d осуществляется при выполнении условия $M = C^d = M^{ed} \bmod p$ для любого $M < p$, для чего степени e и d выбираются такими, чтобы выполнялось условие $ed = 1 \bmod p - 1$. Пара (e,d) составляет локальный ключ отправителя сообщения. Для обеспечения 128-битовой стойкости необходимо использовать в качестве модуля простое число p размером не менее 2 464 бита, причем разложение на множители числа $p-1$ должно содержать, по крайней мере, один большой простой множитель q размером не менее 256 бит.

Уменьшение вычислительной сложности протокола стойкого шифрования по короткому разделяемому ключу K может быть достигнуто при использовании случайного простого числа p в качестве модуля алгоритма коммутативного шифрования и путем шифрования p по ключу K перед его отправкой по открытому каналу (например, отправитель секретного сообщения генерирует случайное простое число p и направляет его получателю в виде криптограммы $\sigma = G_K(p)$). При использовании этого механизма активный нарушитель, выдающий себя за отправителя или получателя сообщения, имеет только одну попытку угадывания секретного ключа K. Если нарушитель выдает себя за отправителя, то это будет обнаружено получателем, так как при расшифровывании им криптограммы σ будет получено число, отличное от p. Если нарушитель выдает себя за получателя, то это будет обнаружено отправителем, так как при расшифровывании криптограммы σ нарушителем будет получено число, отличное от p. Поскольку вмешательство активного нарушителя ведет к получению различных значений модуля, то пропадает свойство коммутативности, и пересылаемое отправителем сообщение не совпадает с сообщением, доставленным получателю в ходе протокола бесключевого шифрования. Последний факт может быть обнаружен с помощью хэш-функции, присоединяемой к сообщению.

При использовании механизма шифрования модуля числа p протокол стойкого шифрования по короткому ключу содержит следующие шаги.

1. Отправитель сообщения M генерирует случайное простое число p (достаточно большого размера), шифрует p по разделяемому ключу K и получает его преобразованное значение $\sigma = G_K(p)$; вычисляет значение h хэш-функции от M; затем, используя алгоритм Полига — Хеллмана, шифрует M по неразделяемому ключу A, получает шифр-текст $C_1 = M^A \bmod p$ и направляет получателю секретного сообщения значения C_1, h и σ по открытому каналу.

2. Получатель расшифровывает шифр-текст σ по разделяемому ключу K, получает значение простого числа p, зашифровывает значение C_1 по неразделяемому ключу B по формуле $C_2 = C_1^B \bmod p$ и направляет отправителю шифр-текст C_2 по открытому каналу.

3. Отправитель расшифровывает шифр-текст C_2 и посылает значение C_3 получателю: $C_3 = C_2^{A^{-1}} \bmod p$.

4. Получатель восстанавливает сообщение M из полученного шифр-текста C_3 по формуле $M = C_3^{B^{-1}} \bmod p$; затем вычисляет значение хэш-функции от M и сравнивает его с h: если сравниваемые значения равны, то получатель делает вывод о подлинности полученного секретного сообщения.

Результаты аутентификации. В обоих вариантах реализации протокола обеспечена неразрывность процедур аутентификации и коммутативного шифрования. Это позволяет избежать активных атак, в которых нарушитель пытается играть роль легального участника протокола. Значения, получаемые в ходе выполнения протокола, вычислительно неотличимы от случайных значений, их шифрование по разделяемому секретному ключу исключает возможность определения короткого ключа методом перебора всех возможных комбинаций разделяемого секретного ключа по ключевому пространству. В этом случае у атакующего отсутствует вычислительно эффективный критерий отбраковки неверных значений ключа. Возможны и другие варианты построения протоколов шифрования с использованием разделяемых ключей малого размера. Представляют интерес конечные поля векторов [5], конечные группы точек эллиптической кривой [6], конечные поля двоичных многочленов, степени которых являются простыми числами Мерсенна [7], также возможна реализация протокола с использованием двух трудных задач [8].

Заключение. Показана принципиальная возможность построения протоколов шифрования по разделяемому секретному ключу малого размера. В основе предложенного способа лежит идея использования бесключевого шифрования совместно с процедурами аутентификации шифр-текстов. В качестве механизма аутентификации используется алгоритм симметричного шифрования по короткому разделяемому ключу некоторых значений, получаемых или используемых в алгоритме коммутативного шифрования. Предложен конкретный вариант реализации протокола на основе алгоритма коммутативного шифрования Полига — Хеллмана. Для обеспечения стойкости протокола, равной 2^{80}, 2^{128} и 2^{160} операциям модульного умножения, следует использовать простое число p, имеющее разрядность 1 024, 2 464 и 4 320 бит соответственно.

Статья подготовлена по результатам работы, выполненной при финансовой поддержке Российского фонда фундаментальных исследований в рамках научного проекта № 14-07-00061-а.

СПИСОК ЛИТЕРАТУРЫ

1. *Schneier B.* Applied Cryptography: Protocols, Algorithms and Source Code. N. Y.: John Wiley & Sons, 1996. 758 p.

2. Pat. 4424414, US. Exponentiation Cryptographic Apparatus and Method / *M. E. Hellman, S. C. Pohlig*. 1984.

3. *Молдовян Н. А.* Введение в криптосистемы с открытым ключом. СПб: БХВ — Петербург, 2007. 286 с.

4. ГОСТ 28147–89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. М.: Изд-во стандартов, 1989. 20 с.

5. *Молдовяну П. А., Молдовян Д. Н., Морозова Е. В., Пилькевич С. В.* Повышение производительности процедур коммутативного шифрования // Вопросы защиты информации. 2009. № 4. С. 24—31.

6. *Молдовян Н. А., Рыжков А. В.* Способ коммутативного шифрования на основе вероятностного кодирования // Вопросы защиты информации. 2013. № 3. С. 3—10.

7. *Демьянчук А. А., Молдовян Н. А., Рыжков А. В.* Выбор „идеальных“ параметров в схеме двухшаговой аутентификации и коммутативном шифре // Изв. СПбГЭТУ „ЛЭТИ“. 2013. № 8. С. 15—18.

8. *Berezin A. N., Moldovyan N. A., Shcherbakov V. A.* Cryptoschemes based on difficulty of simultaneous solving two different difficult problems // Computer Science Journal of Moldova. 2013. Vol. 21, N 2(62). P. 280—290.

Сведения об авторах

Антон Владимирович Муравьев — аспирант; СПИИРАН, научно-исследовательский отдел проблем информационной безопасности; E-mail: muravev.anton@gmail.com

Андрей Николаевич Березин — аспирант; Санкт-Петербургский государственный электротехнический университет „ЛЭТИ“ им. В. И. Ульянова, кафедра автоматизированных систем обработки информации и управления; E-mail: a.n.berezin.ru@gmail.com

Дмитрий Николаевич Молдовян — СПИИРАН, научно-исследовательский отдел проблем информационной безопасности; научный сотрудник; E-mail: mdn.spectr@mail.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

УДК 004.056

Е. В. Дойникова, И. В. Котенко

АНАЛИЗ ТЕКУЩЕЙ СИТУАЦИИ И ПОДДЕРЖКА ПРИНЯТИЯ РЕШЕНИЙ ПО БЕЗОПАСНОСТИ КОМПЬЮТЕРНОЙ СЕТИ НА ОСНОВЕ СИСТЕМЫ ПОКАЗАТЕЛЕЙ ЗАЩИЩЕННОСТИ

Рассматривается подход к отслеживанию текущей ситуации по защищенности компьютерной сети и поддержке принятия решений по реагированию на инциденты, нарушающие информационную безопасность. Подход основан на использовании предлагаемой системы показателей защищенности и разработанных алгоритмов их расчета.

Ключевые слова: *оценивание защищенности, показатели защищенности, графы атак, графы зависимостей сервисов, события информационной безопасности.*

Введение. Сложность архитектуры современных компьютерных сетей и проводимых против них атак, а также многообразие событий, нарушающих информационную безопасность, обусловливает необходимость автоматизированной поддержки принятия решений по реагированию на инциденты (information security incident). Основой для принятия решений по реагированию могут служить показатели защищенности, корректно отражающие текущую ситуацию по безопасности компьютерной сети.

В настоящей статье предлагается система показателей защищенности, приводится ряд известных и модифицированных алгоритмов расчета отдельных и интегральных показателей и рассматривается общий подход к анализу ситуации и принятию решений по безопасности на основе предложенной системы показателей.

Релевантные работы. На данный момент существует большое количество исследований в области применения показателей защищенности для анализа безопасности компьютерных сетей. Однако в большинстве работ анализируются отдельные показатели и не учитываются разные типы информации по безопасности. Так, в работах [1, 2] рассматриваются показатели, рассчитываемые на основе информации о составе и характеристиках объектов ком-

пьютерной сети, например характеристиках хостов сети (критичность хоста или ценность для бизнеса, незащищенность хоста и т.п.), характеристиках сети с позиции приложений (количество приложений, процент критичных приложений и т.п.) и характеристиках сети, учитывающих информацию об уязвимостях (количество систем без известных критичных уязвимостей, количество известных уязвимостей и т.п.).

Показатели, рассчитываемые на основе графов атак, рассматриваются в работах [3, 4]. Данные показатели (такие, как уровень навыков атакующего, атрибуты атакующего, потенциал/вероятность атаки) позволяют получить дополнительную информацию о возможных шагах атакующего в сети с учетом уязвимостей системы. Показатели, рассчитываемые на основе графов зависимостей сервисов, позволяют отследить распространение ущерба в сети (см. работы [4, 5]). В работах [6, 7] рассматриваются показатели, отражающие возможность атак нулевого дня (вероятностная мера уязвимости, k-безопасность нулевого дня).

Для оценивания общего уровня защищенности системы в работах [3, 8] предлагается использовать показатель „уровень риска“, а в работе [9] рассматривается показатель „поверхность атаки“.

Проблемы принятия решений обсуждаются в работах [4, 5, 10], где выделяются показатели, отражающие потери и выигрыш при внедрении ответных мер или отказе от реагирования (например, ожидаемые годовые потери, эффективность реагирования, затраты на реагирование и т.п.).

Ряд работ посвящен созданию различных систем показателей, например, показатели могут быть разделены на первичные и вторичные в зависимости от порядка вычислений [11] или разделены по областям функционирования (управление инцидентами или управление уязвимостями) [1]. В работе [12] выделены восемь категорий показателей в зависимости от типа значения (например, количественное или порядковое).

Подход, рассматриваемый в настоящей статье, базируется на показателях защищенности, предложенных в указанных работах, и подходе к моделированию графа атак, изложенном в работах [13—15]. Основным отличием предлагаемого подхода от других является объединение показателей в комплексную систему, предназначенную для эффективной поддержки принятия решений по реагированию на инциденты, нарушающие информационную безопасность.

Система показателей защищенности. При разработке системы показателей защищенности, алгоритмов их расчета и подхода к оцениванию защищенности сети и поддержке принятия решений учитывался ряд требований, в том числе стандартные требования к показателям, приведенные в работе [16] (такие, как значимость, ценность, объективность, воспроизводимость и т.п.). Основными функциональными требованиями к показателям являются:

— возможность выявления наиболее уязвимых мест системы;

— оценивание потенциала атаки и уровня возможного ущерба в случае ее успешной реализации;

— определение профиля атакующего, его целей и возможностей по реализации атак;

— оценивание выигрыша при реагировании на инциденты;

— учет событий, происходящих в системе, для корректного отображения текущей ситуации.

Подход к оцениванию защищенности и поддержке принятия решений по реагированию на инциденты также должен удовлетворять основным функциональным требованиям, а именно, при реализации подхода должны быть обеспечены:

— всеобъемлющая оценка рисков и помощь администратору по безопасности в принятии решения по реагированию с учетом временных и стоимостных ограничений;

— учет требований стандартов и протоколов в области информационной безопасности.

Основные нефункциональные требования к алгоритмам расчета показателей — оперативность (получение результата за минимальное время) и обоснованность (соответствие результатов оценки реальному состоянию компьютерной сети).

На основе указанных требований была разработана система показателей защищенности, включающая показатели нескольких групп (уровней). В соответствии с информацией, используемой для вычисления показателей, выделены показатели топологического уровня, уровня графа атак, уровня атакующего, уровня событий и интегрального (системного) уровня. На *интегральном уровне* показатели предыдущих уровней используются для определения степени риска и выработки рекомендаций.

Показатели *топологического уровня* основываются на данных о составе и характеристиках сети и позволяют выделить наиболее критичные и уязвимые места системы. К ним относятся: *уязвимость хоста*, *слабость хоста*, *внутренняя критичность*, *внешняя критичность*, *процент систем без известных критичных уязвимостей*, *уязвимость хоста к атакам нулевого дня*, *ценность хоста для бизнеса*. Для учета распространения ущерба в сети через зависимости сервисов на данном уровне строится граф зависимостей сервисов на основе подхода, предложенного в работе [5]. Это позволяет более точно рассчитать критичность хостов сети.

Интегральный показатель риска на данном уровне определяется показателями критичности и уязвимости хостов, которые, однако, не учитывают возможные маршруты атак в сети, т.е. доступность хоста для нарушителя.

На *уровне графа атак* вводится дополнительная информация о связях уязвимостей в сети и строятся возможные маршруты атак, объединенные в граф [13—15]. На данном уровне определяются такие показатели, как: *критичность атакующих действий*, *потенциал атаки*, *ущерб от атаки*, *потенциал атаки с учетом нулевого дня*, *стоимостный ущерб от атаки*, *затраты на реагирование*. Для определения вероятностей атаки граф атак преобразуется в граф уязвимостей, вершины которого определяют соответствующие уязвимости, а дуги — переходы между ними. При этом для уменьшения объема вычислений уязвимости делятся на группы в соответствии с индексами Общей системы оценивания уязвимостей (Common Vulnerability Scoring System — CVSS): вектор доступа, сложность доступа и аутентификация [17]. При успешной реализации уязвимости, относящейся к какой-либо группе, атакующий может нанести ущерб хосту или расширить свои права доступа к системе. Вероятность успешной реализации уязвимостей группы определяется как произведение значений указанных индексов CVSS. Для учета успешной реализации уязвимостей, необходимых для достижения предусловий реализации текущей уязвимости, используется формула определения совместной вероятности. Таким образом, при определении интегрального показателя риска на данном уровне, кроме критичности хоста, учитывается вероятность успешной реализации атаки на хост.

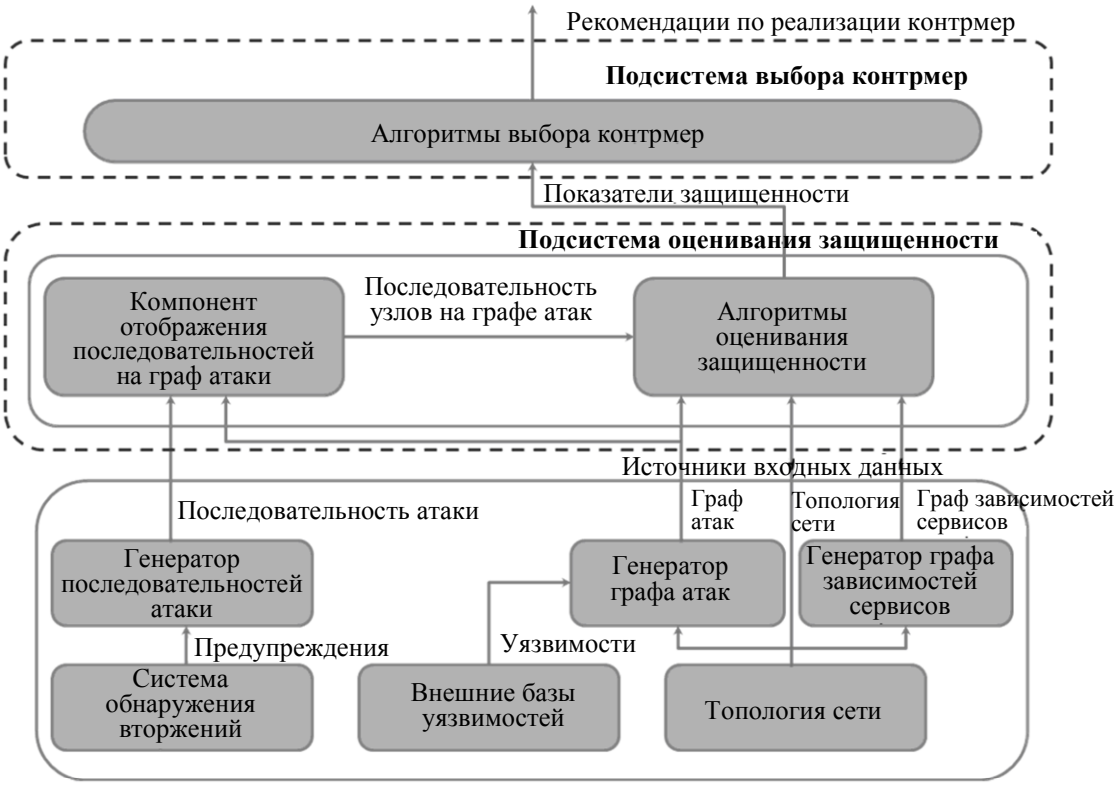
На *уровне атакующего* вводится информация, связанная с различными моделями, характеризующими атакующего, в том числе положение атакующего в системе (внутренний или внешний), уровень навыков атакующего (высокий, средний или низкий) и цель атакующего. На данном уровне определяются следующие показатели: *уровень навыков атакующего*, *профильный потенциал атаки*, *профильный потенциал атаки с учетом нулевого дня*, *профильный стоимостный ущерб от атаки*, *профильные затраты на реагирование*. Данный уровень позволяет при определении интегрального показателя риска учитывать заданные модели.

На *уровне событий* при вычислении показателей учитывается информация о событиях в сети (атакованном хосте и привилегиях, полученных атакующим). Показателями данного уровня являются: *позиция атакующего*, *динамический уровень навыков атакующего*, *вероятностный уровень навыков атакующего*, *динамический потенциал атаки*, *динамический потенциал атаки с учетом нулевого дня*, *динамический стоимостный ущерб от атаки*, *динамические затраты на реагирование*. Данный уровень относится к динамическому режиму функционирования системы и позволяет отражать текущую ситуацию по защищенности

в виде профиля атаки и профиля атакующего. Профиль атаки содержит информацию о текущей позиции атаки на графе атак (в соответствии с поступившими нарушающими безопасность событиями), о наиболее вероятных предыдущих шагах атаки (определяемых на основе теоремы Байеса об апостериорных вероятностях), о наиболее вероятных будущих шагах атаки и цели атакующего. Профиль атакующего содержит информацию о проводимой атаке и наиболее вероятном уровне навыков атакующего. Таким образом, данный уровень позволяет при определении интегрального показателя риска учитывать информацию о развитии проходящей в сети атаки.

Принятие решений по реагированию основывается на учете возможных контрмер для каждой уязвимости графа и решении оптимизационной задачи с использованием предложенных показателей (т.е. минимизации таких показателей, как ущерб и затраты на реагирование при большом показателе вероятности атаки).

Архитектура системы оценивания защищенности и поддержки принятия решений. Предложенные алгоритмы были реализованы в рамках архитектуры, представленной на рисунке.



Система включает две основные подсистемы: 1) подсистему оценивания защищенности и 2) подсистему выбора контрмер. 2-я подсистема генерирует рекомендации по реализации контрмер на основе показателей, полученных от подсистемы оценивания защищенности. 1-я подсистема содержит набор алгоритмов оценивания защищенности и компонент отображения последовательностей событий на граф атак. Подсистема оценивания защищенности получает входные данные от генератора графа атак, генератора графа зависимостей сервисов и генератора последовательностей атак.

Заключение. Представленный подход к отслеживанию текущей ситуации по защищенности компьютерной сети и поддержке принятия решений по выработке контрмер основан на системе показателей защищенности. Описаны основные показатели, соответствующие разным уровням системы, и алгоритмы их расчета. На основе предложенного подхода разработан прототип системы оценивания защищенности и поддержки принятия решений, позволяющий

отследить ситуацию по безопасности в информационной системе и выбрать оптимальный набор контрмер с использованием системы показателей.

Статья подготовлена по результатам работы, выполняемой при финансовой поддержке Российского фонда фундаментальных исследований (гранты 13-01-00843, 13-07-13159, 14-07-00697, 14-07-00417, 14-37-50735), программы фундаментальных исследований ОНИТ РАН (контракт № 2.2), проекта ENGENSEC программы Европейского сообщества TEMPUS и государственных контрактов № 14.604.21.0033, 14.604.21.0137, 14.604.21.0147 и 14.616.21.0028.

СПИСОК ЛИТЕРАТУРЫ

1. The Center for Internet Security. The CIS Security Metrics, v.1.0.0, 2009 [Электронный ресурс]: <https://buildsecurityin.us-cert.gov/sites/default/files/CIS_Security_Metrics_v1.0.0.pdf>, 11.2014.
2. *Mayer A*. Operational Security Risk Metrics: Definitions, Calculations, Visualizations // Metricon 2.0. CTO RedSeal Systems, 2007.
3. *Dantu R., Kolan P., Cangussu J.* Network risk management using attacker profiling // Security and Communication Networks. 2009. Vol. 2, N 1. P. 83—96.
4. *Kanoun W., Cuppens-Boulahia N., Cuppens F., Araujo J.* Automated reaction based on risk analysis and attackers skills in intrusion detection systems // Proc. of the 3rd Intern. Conf. on Risks and Security of Internet and Systems (CRiSIS'08). Toezer, Tunisia, 2008. P. 117—124.
5. *Kheir N., Cuppens-Boulahia N., Cuppens F., Debar H.* A service dependency model for cost-sensitive intrusion response // Proc. of the 15th European Symp. on Research in Computer Security (ESORICS'10). Athens, Greece, 2010. P. 626—642.
6. *Ahmed M. S., Al-Shaer E., Khan L.* A novel quantitative approach for measuring network security // Proc. of the 27th Conf. on Computer Communications (INFOCOM'08). Phoenix, Arizona, 2008. P. 1957—1965.
7. *Wang L., Singhal A., Jajodia S., Noel S.* k-zero day safety: measuring the security risk of networks against unknown attacks // Proc. of the 15th European Conf. on Research in Computer Security. Berlin, Heidelberg: Springer-Verlag, 2010. P. 573—587.
8. *Kotenko I., Saenko I., Polubelova O., Doynikova E.* The ontology of metrics for security evaluation and decision support in SIEM systems // IEEE 2nd Intern. Workshop on Recent Advances in Security Information and Event Management (RaSIEM 2013); In conjunction with ARES 2013. Regensburg, Germany, 2013. P. 638—645.
9. *Manadhata P. K., Wing J. M.* An attack surface metric // IEEE Transact. on Software Engineering, 2010. P. 371—386.
10. *Jahnke M., Thul C., Martini P.* Graph-based metrics for intrusion response measures in computer networks // IEEE Workshop on Network Security. 2007. P. 1035—1042.
11. *Idika N. C.* Characterizing and Aggregating Attack Graph-Based Security Metric: PhD Thesis, Purdue University, 2010. P. 1—131.
12. *Axelrod C. W.* Accounting for value and uncertainty in security metrics // Information Systems Control J. 2008. Vol. 6. P. 1—6.
13. *Kotenko I., Stepashkin M.* Network security evaluation based on simulation of malefactor's behavior // Proc. of the Intern. Conf. on Security and Cryptography (SECRYPT'06). Setubal, Portugal, 2006. P. 339—344.
14. *Котенко И. В., Степашкин М. В., Богданов В. С.* Архитектуры и модели компонентов активного анализа защищенности на основе имитации действий злоумышленников // Проблемы информационной безопасности. Компьютерные системы. 2006. № 2. С. 7—24.
15. *Kotenko I., Chechulin A.* Computer attack modeling and security evaluation based on attack graphs // IEEE 7th Intern. Conf. on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS'2013). Berlin, Germany, 12—14 Sept., 2013. P. 614—619.
16. *Barabanov R.* Information security metrics. State of the art // DSV Report Ser. 2011. N 11—007, March.
17. Common Vulnerability Scoring System (CVSS) [Электронный ресурс]: <http: // www.first.org / cvss>, 09.2010.

Сведения об авторах

Елена Владимировна Дойникова — аспирант; СПИИРАН, лаборатория проблем компьютерной безопасности; E-mail: doynikova@comsec.spb.ru

Игорь Витальевич Котенко — д-р техн. наук, профессор; СПИИРАН, лаборатория проблем компьютерной безопасности; E-mail: ivkote@comsec.spb.ru

Рекомендована СПИИРАН

Поступила в редакцию
10.06.14 г.

SUMMARY

P. 7—15.

CONCEPT OF PROACTIVE CONTROL OVER COMPLEX OBJECTS: THEORETICAL AND TECHNOLOGICAL BASIS

Theoretical and technological background of applied theory of proactive control over complex objects is considered. The theory under development has been actually realized in applications to space-rocket industry, atomic-power engineering, transport and logistics, and military technology.

Keywords: interdisciplinary approach, complexity management, proactive monitoring and control, complex modeling.

Data on authors

Mikhail Yu. Okhtilev — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: oxt@mail.ru

Nikolay G. Mustafin — Cand. Techn. Sci.; St. Petersburg State Electrotechnical University "LETI", Department of Automated Systems of Information Processing and Control; E-mail: nikolay.mustafin@gmail.com

Vladimir E. Miller — Cand. Techn. Sci.; Joint Stock Company "Academician A. L. Mints Radiotechnical Institute", St. Petersburg; Subdivision Director; E-mail: miller@progsystema.ru

Boris V. Sokolov — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; Deputy Director for R&D; E-mail: sokol@iias.spb.su

P. 15—18.

OPTIMIZATION OF THE ROUTING MATRIX IN QUEUING NETWORKS

An algorithm for calculation of temporal characteristics of open queuing network is described. The method is proposed to optimize queuing network demand sojourn time by equalizing the nodes loading. Results of the computer experiment are presented and discussed.

Keywords: open networks, sojourn time, nodes loading equalizing.

Data on author

Yury I. Ryzhikov — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; Mozhaisky Military Space Academy, St. Petersburg, Department of Software Engineering; E-mail: ryzhbox@yandex.ru

P. 19—25.

INFORMATION MODEL OF DISTRIBUTED EVENTS SUPPORT AT INTELLIGENT MEETING ROOM

The problem of information-technical support of distributed events is considered. The main arrangement stages of event with distributed participants are analyzed. An information model is proposed to describe the methods of data processing and exchange between the distributed participants for various situations in the intelligent meeting room.

Keywords: intelligent environment, distributed meeting, audiovisual data processing, speaker diarization, information significance.

Data on authors

Viktor Yu. Budkov — Cand. Techn. Sci.; St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Speech and Multimodal Interfaces; Scientist; E-mail: budkov@iias.spb.su

Andrey L. Ronzhin — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Speech and Multimodal Interfaces; Deputy Director for R&D; E-mail: ronzhin@iias.spb.su

P. 25—30.

MODIFIED MODEL OF FLEXIBLE REDISTRIBUTION OF INFORMATION INTERACTION OPERATIONS

A modified mathematical model of planning of decentralized information processing in dynamically changing environment is proposed. The model accounts for restrictions on implicit time for execution of operations of information processing, storage, and transmission in space systems.

Keywords: dynamic network, communication, implicit time restriction.

Data on authors

Alexander N. Pavlov — Cand. Techn. Sci.; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: pavlov62@list.ru

Dmitry A. Pavlov — Post-Graduate Student; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: dpavlov239@mail.ru

Boris V. Moskvin — Cand. Techn. Sci., Professor; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: z-moskvin@mail.ru

Kirill L. Grigoriev — Cand. Techn. Sci.; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: Grigorjev.kir@yandex.ru

P. 30—34.

VALUATION MODEL OF SPACECRAFT FUEL CONSUMPTION WITH CONSIDERATION FOR CONTINGENCY

A simplified functional model of spacecraft functioning is proposed for evaluation of its fuel resources and active life time. Effects of on-board failures of motion control system on fuel consumption at normal conditions as well as in an emergency are taken into account.

Keywords: modeling of failures, fuel resources, emergency situation, on-board spacecraft system.

Data on author

Alexander Yu. Kulakov — Post-Graduate Student; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: russ69@bk.ru

P. 35—40.

MODELING AND IDENTIFICATION OF OBJECT-ORIENTED SOFTWARE CODE DEFECTS

An approach to the problem of modeling and identification of program code defects aimed at improvement of the software quality is presented. The approach is based on the graph representation of the application source code and its complex analysis.

Keywords: software modeling, graph model, program defect.

Data on author

Vadim V. Burakov — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: Burakov@eureca.ru

P. 40—45.

METHOD OF CF-GRAMMAR REGULARIZATION FOR LANGUAGE PROCESSORS

A method of context-free regularization based on special equivalent transformations of the grammar syntactic graph is described. In combination with an algorithm of elimination of recursions, the method is shown to ultimately convert a context-free grammar into a regular one.

Keywords: context-free grammar, equivalent transformations of grammars.

Data on author

Ludmila N. Fedorchenko — Cand. Techn. Sci.; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Applied Informatics; Senior Scientist; E-mail: lnf@iias.spb.su

P. 46—52.

SYNTHESIS OF STRUCTURAL DYNAMICS MODELING SCENARIOS FOR AUTOMATED CONTROL SYSTEM OF ACTIVE MOVING OBJECTS

Feasible technologies of synthesis of structural dynamics simulation scenarios for automated control system of active moving objects are analyzed. Service-oriented approach is proposed for practical realization of the scenarios; the corresponding mathematical software is considered.

Keywords: complex modeling, simulation scenarios synthesis technologies, service-oriented approach.

Data on author

Semyon A. Potryasaev — Cand. Techn. Sci.; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: spotryasaev@gmail.com

P. 52—57.

STUDY OF DYNAMIC AND ENERGY COMPATIBILITY OF POSITIONING SYSTEM AND ANGULAR MOTION CONTROL OF SPACE SOLAR POWER PLANT

Interaction between positioning system and angular motion control of space solar power plant is studied within the framework of optimization problems of structural-parametrical synthesis of the plant main subsystems.

Keywords: space solar energy station, the positioning system and management of angular motion, solar panel bat-Rey, solar concentrators.

Data on authors

Yury S. Manuilov — Dr. Techn. Sci., Professor; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: ymanoff@yahoo.com, kotmanoff@rambler.ru

Sergey V. Zinoviev — Cand. Techn. Sci.; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: zinoviev_sv@mail.ru

Yury V. Prishchepa — Cand. Techn. Sci.; Public Corporation «Concern Radiostroeniya "Vega"», St. Petersburg Branch; Director; E-mail: mail@spb.vega.su

Evgeny N. Aleshin — Cand. Techn. Sci.; Mozhaysky Military Space Academy, Department of Automated Control Systems of Space Complexes, St. Petersburg; E-mail: aleshin_evgeny@inbox.ru

P. 58—62.

THE SAMPO+ SYSTEM FOR CREATION AND ANALYSIS OF SOFTWARE HISTORICAL DATABASES

A methodology for development of historical databases designed for accurate estimation of necessary resources for initiated software projects is proposed. A procedure of the database creation with the use of SAMPO+ system is presented.

Keywords: project historical databases, collecting and analyzing project metrics.

Data on author

Alexander M. Telezhkin — Post-Graduate Student; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Information Technologies in System Analysis and Modeling; E-mail: telezhkin@gmail.com

P. 62—67.

CONSTRUCTION OF INTEGRATED BASE OF VULNERABILITIES

Results of analysis of open databases of vulnerabilities are presented. The process of the bases integration in evaluation system of computer networks security is described. A model of the base formation process and structure of integrated base of vulnerabilities is proposed.

Keywords: security evaluation, database of vulnerabilities, security monitoring system.

Data on authors

Andrey V. Fedorchenko — St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Computer Security Problems; Junior Scientist; E-mail: fedorchenko@comsec.spb.ru

Andrey A. Chechulin — Cand. Techn. Sci.; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Computer Security Problems; E-mail: chechulin@comsec.spb.ru

Igor V. Kotenko — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Computer Security Problems; E-mail: ivkote@comsec.spb.ru

P. 68—72.

SECURE ENCRYPTION PROTOCOL EMPLOYING SHORT KEYS

A method and protocol for secure cryptographic transformation of information transmitted via public channels are proposed. Small-sized (up to 56 bit) shared secret keys are employed.

Keywords: encryption, cryptographic protocol, secret key, security, discrete logarithm problem, commutative encryption.

Data on authors

Anton V. Muravev — Post-Graduate Student; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Department of Information Security Problems;
E-mail: muravev.anton@gmail.com

Andrey N. Berezin — Post-Graduate Student; St. Petersburg State Electrotechnical University "LETI", Department of Automated Information Processing and Control Systems;
E-mail: a.n.berezin.ru@gmail.com

Dmitry N. Moldovyan — St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Department of Information Security Problems; Scientist;
E-mail: mdn.spectr@mail.ru

P. 72—77.

MONITORING OF CURRENT SITUATION AND SUPPORT OF DECISION MAKING IN COMPUTER NETWORK SECURITY BASED ON THE SECURITY METRICS SYSTEM

An approach to monitoring of current security situation and support of decision on response to security deterioration incidents is proposed. The approach is based on developed system of security characteristics and models and algorithms for evaluation of the characteristics.

Keywords: security assessment, security metrics, attack graphs, service dependencies graphs, information security events.

Data on authors

Elena V. Doynikova — Post-Graduate Student; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Computer Security Problems;
E-mail: doynikova@comsec.spb.ru

Igor V. Kotenko — Dr. Techn. Sci., Professor; St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Laboratory of Computer Security Problems;
E-mail: ivkote@comsec.spb.ru