А.Г. Сухарев

ОПТИМИЗАЦИЯ
И ИССЛЕДОВАНИЕ
ОПЕРАЦИЙ

МИНИМАКСНЫЕ АЛГОРИТМЫ В ЗАДАЧАХ ЧИСЛЕННОГО АНАЛИЗА

ОПТИМИЗАЦИЯ И ИССЛЕДОВАНИЕ
ОПЕРАЦИЙ

А. Г. СУХАРЕВ

МИНИМАКСНЫЕ АЛГОРИТМЫ В ЗАДАЧАХ ЧИСЛЕННОГО АНАЛИЗА

МОСКВА «НАУКА»
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1989

ББК 22.193
С91
УДК 519.6

ОПТИМИЗАЦИЯ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

ВЫПУСК 22

Редактор серии *Н. Н. МОИСЕЕВ*

Сухарев А. Г. **Минимаксные алгоритмы в задачах численного анализа.**— М.: Наука. Гл. ред. физ.-мат. лит., 1989.— 304 с.— (Оптимизация и исследование операций/Ред. сер. Н. Н. Моисеев).— ISBN 5-02-013942-4.

В монографии исследуются вопросы эффективности и оптимальности алгоритмов решения задач численного анализа. Рассмотрение ведется в рамках общей модели, в основу которой положена минимаксная концепция оптимальности вычислительных алгоритмов. При построении оптимальных алгоритмов широко используются теоретико-игровые и другие методы исследования операций и системного анализа.

Для студентов, аспирантов и специалистов в области прикладной математики.

Табл. 5. Ил. 18. Библиогр. 468 назв.

Рецензент
доктор физико-математических наук *В. И. Благодатских*

$\frac{1602120000-113}{053(02)-89}$ 28-89

© Издательство «Наука». Главная редакция физико-математической литературы, 1989

ISBN 5-02-013942-4

ОГЛАВЛЕНИЕ

Предисловие 5

Список обозначений 8

Глава 1. **Общая модель вычислений** 9

§ 1. Основные понятия 9
§ 2. О рассматриваемых функциональных классах . . 12
§ 3. Классы детерминированных алгоритмов 21
§ 4. Минимаксная концепция оптимальности вычислительных алгоритмов и конкретные понятия оптимальности 28
§ 5. Сравнение наилучших гарантированных результатов в классах пассивных и последовательных алгоритмов 49
§ 6. Последовательно-оптимальные алгоритмы . . . 56
§ 7. Стохастические алгоритмы 62

Глава 2. **Вычисление интегралов** 73

§ 1. Оптимальные квадратуры для функциональных классов, заданных квазиметриками 73
§ 2. Построение и исследование оптимальных квадратур для функциональных классов, заданных модулями непрерывности 90
§ 3. Последовательно-оптимальные и оптимальные на один шаг алгоритмы численного интегрирования 100
§ 4. Численный эксперимент 124
§ 5. Оптимальное вычисление повторных интегралов . . 132
§ 6. Вычисление кратных интегралов с помощью разверток типа кривой Пеано 147

Глава 3. **Восстановление функций по их значениям** . . 153

§ 1. Оптимальные пассивные алгоритмы 153
§ 2. Последовательно-оптимальные и оптимальные на один шаг алгоритмы восстановления функций . . 161
§ 3. Решение многошаговой антагонистической игры, связанной с задачей оптимального восстановления 173

Глава 4. **Поиск глобального экстремума** 191

§ 1. О выборе начальных точек для методов локальной оптимизации 192
§ 2. Оптимальный пассивный поиск для функционального класса, заданного квазиметрикой 198
§ 3. Сведéние задачи построения последовательно-оптимального алгоритма для функционального класса,

заданного квазиметрикой, к серии задач оптимального покрытия 205
§ 4. Конкретные вычислительные алгоритмы 212
§ 5. Учет погрешности вычислений 222
§ 6. Оптимальный на один шаг стохастический алгоритм 228

Глава 5. **Некоторые специальные классы экстремальных задач** 244
§ 1. Решение уравнений и систем уравнений . . . 244
§ 2. Отыскание максимума функции минимума при связанных переменных 256
§ 3. Многокритериальные задачи оптимизации . . . 261

Список литературы 270

Предметный указатель 295

Посвящается
памяти моей матери
Александры Ильиничны Сухаревой
и моего отца
Григория Михайловича Сухарева

ПРЕДИСЛОВИЕ

Вопросы эффективности численных методов, выбора наиболее эффективных методов для решения данной задачи или данного класса задач всегда занимали важное место в численном анализе. И в настоящее время оптимизация вычислительного процесса решения задач на ЭВМ является актуальной проблемой вычислительной математики, проблемой, которая стимулирует поиск новых численных методов и способов их реализации.

В предлагаемой книге для различных задач численного анализа изучаются способы оценки эффективности вычислительных алгоритмов и вопросы их оптимальности в рамках общей модели вычислений. Основными элементами общей модели, определяющими конкретные модели вычислений, являются: приближаемый функционал или оператор, соответствующий решаемой задаче; класс функций, с помощью которого описывается информация вычислителя о задаче; класс алгоритмов решения задачи, находящихся в распоряжении вычислителя; критерий оценки эффективности алгоритмов; концепция оптимальности; и, наконец, конкретное понятие оптимальности алгоритма в рамках принятой концепции. Использование общей модели вычислений позволяет дать единую трактовку различных задач численного анализа, ответить на ряд принципиальных методологических вопросов и установить общие свойства оптимальных алгоритмов.

В основу всех рассматриваемых в книге конкретных реализаций общей модели вычислений положена минимаксная концепция оптимальности, которую использовал в задачах вычислительной математики еще П. Л. Чебышев. Эта концепция отражает стремление вычислителя к получению наилучшего гарантированного результата (например, точности решения) относительно находящейся в распоряжении вычислителя информации о задаче.

Оптимальным алгоритмам решения задач численного анализа посвящена достаточно обширная журнальная литература. Имеются также несколько монографий, где наряду с рассмотрением конкретных задач закладываются основы общей теории оптимальных алгоритмов.

Тематика данной книги существенно отличается от традиционной тематики оптимальных численных методов, что выражается прежде всего в углубленном внимании к последовательным (адаптивным) вычислительным алгоритмам. При этом процесс вычислений рассматривается как управляемый процесс, а сам алгоритм — как стратегия управления. Такой подход позволяет при построении оптимальных алгоритмов широко использовать теоретико-игровые и другие методы исследования операций и системного анализа. Использование этих методов оказывается плодотворным и позволяет получить целый ряд результатов, относящихся как к традиционным, так и к новым понятиям оптимальности.

Конечной целью исследования различных изучаемых в книге моделей вычислений является построение конкретного вычислительного алгоритма, допускающего программную реализацию. Центральное место занимает понятие последовательно-оптимального алгоритма, отражающее во многих случаях особенности организации реальных вычислительных процессов полнее, чем традиционные понятия оптимальности. Разработана общая схема построения последовательно-оптимальных алгоритмов. Сформулированы требования к модели вычислений, обеспечивающие возможность применения этой схемы для ряда задач численного анализа.

Книга состоит из пяти глав. Первая глава стоит особняком от остальных четырех. В ней вводится необходимая терминология, строится общая модель вычислений, подробно обсуждаются возникающие при этом методологические вопросы. Следует подчеркнуть, что относящуюся к оптимальным алгоритмам терминологию нельзя считать полностью установившейся. Так, вводимый в § 3 гл. 1 термин «итоговая операция алгоритма» соответствует термину «алгоритм» монографии Дж. Трауба, Х. Вожьняковского (см. список литературы). Полученные в первой главе результаты неоднократно используются в дальнейшем. Она, таким образом, носит в известной степени вводный характер.

Последующие главы посвящены построению оптимальных методов решения конкретных задач численного анализа. В книге рассматриваются задачи вычисления интегралов, восстановления функций по их значениям, поиска глобального экстремума, решения уравнений и систем уравнений, отыскания максимина и многокритериальные задачи оптимизации. Разумеется, каждая задача исследуется в рамках конкретных, порою довольно ограничительных, предположений о функциональных классах и классах алгоритмов, о критериях оценки эффективности алгоритмов и используемых понятиях оптимальности, т. е. в рамках конкретной модели вычислений. Выбор в книге конкретных моделей определяется желанием наиболее полно продемонстрировать возможности, заложенные в общей модели, а также стремлением полнее учесть особенности реальных задач. Некоторые результаты, тесно связанные с тематикой книги (например, результаты по оптимальному поиску экстремума унимодальных функций), не вошли в нее лишь потому, что они уже неоднократно излагались и в монографиях, и в учебной литературе.

Список литературы достаточно обширен и включает многие работы, на которые нет непосредственных ссылок в тексте книги. В них либо изучаются те или иные вопросы оптимальности или оценки эффективности численных методов, либо эти работы тесно связаны с рассматриваемыми в книге конкретными задачами численного анализа. В то же время в список литературы в целях экономии места не вошел ряд имеющих непосредственное отношение к тематике книги работ из аннотированной библиографии к монографии Дж. Трауба, Х. Вожьняковского.

Несколько слов о нумерации теорем, лемм и формул. В пределах каждой главы принята двойная нумерация: первое число указывает номер параграфа, второе — номер соответствующей теоремы, леммы или формулы. При ссылках в пределах данной главы указываются только эти два номера. В противном случае к ним добавляется номер главы, например: теорема 5.3 гл. 1, формула (3.4) гл. 2.

Автор глубоко признателен всему коллективу кафедры исследования операций факультета вычислительной математики и кибернетики МГУ, творческая атмосфера которого имела огромное значение для создания этой книги.

СПИСОК ОБОЗНАЧЕНИЙ

Здесь даны лишь некоторые обозначения общего характера. Не приводятся многие общеупотребительные обозначения, а также специальные обозначения, вводимые в книге. Эти последние либо используются непосредственно после того, как они введены, либо сопровождаются в необходимых местах соответствующими ссылками.

$\overset{\text{def}}{=}$ — равно по определению (при этом определяемая величина может стоять как слева, так и справа от знака $\overset{\text{def}}{=}$);

$\Rightarrow$ — знак импликации: «из... следует...»;

$\Leftrightarrow$ — знак эквивалентности;

$A \subset B$ — множество A содержится в множестве B (это не исключает случая $A = B$);

$\lfloor t \rfloor$ — целая часть t, т. е. наибольшее целое число, меньшее или равное t;

$\lceil t \rceil$ — наименьшее целое число, большее или равное t;

$x := a$ — оператор, присваивающий переменной x значение a;

$\triangle$ — символ конца доказательства, конца формулировки утверждения, если оно следует из предшествующих рассуждений или приводится без доказательства, а также конца формулировки для выделенных соответствующими заголовками замечаний, примеров, определений и т. п.;

$\mathbf{R}$ — числовая прямая (множество вещественных чисел);

$\mathbf{R}^n$ — n-мерное координатное пространство;

$A + B = \{c \mid c = a + b,\ a \in A,\ b \in B\}$;

$\langle a, b \rangle = \sum_{j=1}^{n} a^j b^j$ — скалярное произведение векторов $a = (a^1, \ldots, a^n)$ и $b = (b^1, \ldots, b^n)$;

$\| a \|_0 = \max_{i=1,\ldots,n} | a^i |$;

$\| a \|_1 = \sum_{i=1}^{n} | a^i |$;

$\| a \|_2 = \sqrt{\sum_{i=1}^{n} (a^i)^2}$;

$\lim_{\delta \to a+} \varphi(\delta)$, $\lim_{\delta \to a-} \varphi(\delta)$ — правосторонний и левосторонний пределы функции φ числового аргумента в точке a;

$x_* = \arg\max_{x \in K} f(x)$ — любая точка глобального максимума функции f на множестве K: $f(x_*) = \max_{x \in K} f(x)$, $x_* \in K$;

$\operatorname{Arg}\max_{x \in K} f(x)$ — множество всех точек глобального максимума функции f на множестве K.

ГЛАВА 1

ОБЩАЯ МОДЕЛЬ ВЫЧИСЛЕНИЙ

В данной главе обсуждаются основные элементы всех дальнейших построений: приближаемый функционал или оператор, соответствующий решаемой задаче; класс функций, с помощью которого описывается информация вычислителя о задаче; класс алгоритмов решения задачи, находящихся в распоряжении вычислителя; критерий оценки эффективности алгоритмов; концепция оптимальности; и, наконец, конкретное понятие оптимальности алгоритма в рамках принятой концепции. Совокупность этих элементов и определяет то, что мы называем *моделью вычислений*. Получены некоторые общие результаты, позволяющие дать ответ на ряд принципиальных методологических вопросов и неоднократно используемые в дальнейших построениях.

§ 1. Основные понятия

В этом параграфе схематически описываются некоторые элементы общей модели вычислений. Показано, как укладываются в данную схему некоторые конкретные задачи численного анализа. В последующих параграфах главы заканчивается описание общей модели и дается необходимая конкретизация введенных понятий.

1. Схема общей модели вычислений. Пусть F — множество в некотором линейном пространстве над полем вещественных чисел, S — отображение, действующее из F в некоторое метрическое пространство B с метрикой γ. Перед вычислителем стоит задача построения «наилучшего приближения» к элементу $S(f)\in B$, т. е. такого приближения $\alpha(f)\in B$, что величина

$$\varepsilon(\alpha,\ f)=\gamma(S(f),\ \alpha(f)) \tag{1.1}$$

«минимальна» (точный смысл терминов, взятых в кавычки, будет определен позднее).

Вычислителем мы здесь и далее именуем лицо, которое строит модель вычислений, организует проведение

вычислений в рамках построенной модели и анализирует полученные результаты.

Величину $\varepsilon(\alpha, f)$ будем называть *погрешностью* (или *точностью*) решения. Она выступает в дальнейшем в роли *критерия оценки эффективности алгоритмов* (будут рассмотрены и другие, отличные от (1.1), критерии оценки эффективности).

Априорная (выявленная в результате предварительного исследования до начала вычислений) информация вычислителя о элементе f состоит в знании того факта, что $f \in F$. В процессе решения задачи с помощью *алгоритма* α вычислитель получает дополнительную информацию, на основании которой строит приближение $\alpha(f)$ (подробнее об этом — в § 3).

2. Примеры. Прежде чем приступить к уточнению описанной схемы общей модели вычислений, поясним ее на конкретных примерах.

Пусть F — класс вещественных скалярных или векторных функций, определенных на множестве K евклидова пространства $\mathbf{R}^n$. Этот случай, как правило, и рассматривается в дальнейшем.

Задаче *численного интегрирования* (вычисления интеграла) в нашей общей схеме соответствует

$$S(f) = \int_K f(x)\,dx, \quad B = \mathbf{R}. \tag{1.2}$$

Задаче *восстановления* функции f соответствует

$$S(f) = f, \quad B \subset \{\beta \mid \beta\colon K \to \mathbf{R}\}. \tag{1.3}$$

Задаче *глобальной оптимизации* (вычисления максимума или верхней грани скалярной функции f на множестве K) —

$$S(f) = \sup_{x \in K} f(x), \quad B = \mathbf{R}, \tag{1.4}$$

или же

$$S(f) = x_*, \quad B = \mathbf{R}^n,$$

где $x_* \in K$, $f(x_*) = \max\limits_{x \in K} f(x)$ (при этом предполагается, что любая функция $f \in F$ достигает своего максимального значения в единственной точке).

Наконец, задаче *решения системы уравнений*

$$f(x) = 0 \tag{1.5}$$

соответствует

$$S(f) = x_*, \qquad B = \mathbf{R}^n, \tag{1.6}$$

где x_* — решение системы. При этом предполагается, что F — класс таких n-мерных вектор-функций f, что система (1.5) имеет единственное решение.

В дальнейшем мы уделим наибольшее внимание задачам (1.2)—(1.4) и задаче решения системы (1.5) в постановке, несколько отличающейся от (1.6). Следует, однако, подчеркнуть, что определения гл. 1 и производимые построения применимы и к другим задачам численного анализа.

Отметим, что термин «задача» в соответствии со сложившимися традициями используется нами для обозначения разных объектов. Так, мы называем задачей отображение S (задача оптимизации, интегрирования и т. д.), совокупность S и класса F, а также конкретный элемент $f \in F$ (в последнем случае подразумевается, что речь идет о каком-то определенном S).

3. Априорная информация о f. Обсудим теперь подробнее методологические вопросы, связанные с целесообразностью фиксации в модели вычислений определенного функционального класса F. Как правило, перед вычислителем стоит одна задача или серия вполне конкретных задач (здесь мы предполагаем, что S, B фиксированы, и задачами именуем конкретные функции f). Тем самым, казалось бы, решен вопрос о выборе класса F — он состоит из единственной функции или совокупности всех рассматриваемых функций.

Однако такое определение F не поможет в решении вопроса о выборе вычислительного алгоритма. Даже если вычислитель имеет дело с одной-единственной функцией f, при выборе алгоритма он может руководствоваться лишь некоторыми известными ему и существенными для него свойствами этой функции. Такими свойствами могут быть непрерывность, гладкость, монотонность, выпуклость, унимодальность (наличие единственного экстремума), ограниченность «скорости изменения» самой функции или ее производных и т. д.

Подобные свойства можно выявить, например, изучая физическую сущность явления при построении его математической модели. Могут быть точно или приближенно известны (или вычислены в результате предварительного исследования) значения некоторых функци-

ональных характеристик. Может быть известно хорошее начальное приближение, позволяющее получить решение с помощью некоторого итерационного процесса. Информация такого рода позволяет вычислителю отнести рассматриваемую функцию или совокупность рассматриваемых функций к тому или иному функциональному классу.

Таким образом, фиксация функционального класса является завершением этапа предварительного исследования — этапа, где выявляются те свойства задачи, которые будут использованы при выборе алгоритма ее решения.

Конечно, более тщательное предварительное исследование, быть может, позволило бы отнести функцию к более узкому классу. Поэтому существует естественное противоречие между стремлением, с одной стороны, получше описать задачу, учесть при выборе алгоритма решения большее число свойств задачи (тем самым усложняя вопрос о выборе алгоритма решения), т. е. стремлением выбрать достаточно узкий класс, и, с другой стороны, стремлением выбрать класс, для которого удалось бы в приемлемое время решить задачу построения эффективного вычислительного алгоритма. Разумный компромисс основывается лишь на квалификации, опыте и интуиции исследователя.

Подчеркнем еще раз, что после фиксации класса F вся используемая вычислителем при выборе алгоритма решения информация о задаче состоит в знании того факта, что $f \in F$.

§ 2. О рассматриваемых функциональных классах

Цель данного параграфа — дать пример важных, с точки зрения теории и приложений, функциональных классов, адекватно описывающих типичную для ряда ситуаций априорную информацию исследователя о задаче. Таким примером являются функциональные классы, заданные квазиметриками. Эти классы фигурируют во многих конкретных моделях вычислений, рассматриваемых в книге. Здесь обосновывается целесообразность использования функциональных классов, заданных квазиметриками, и изучаются их важнейшие свойства.

1. Неформальное и формальное описание априорной информации, типичной для ряда задач. В вычислительной практике встречается большое число задач с функ-

циями весьма сложной структуры. Для вычисления каждой функциональной характеристики (например, значения функции) может потребоваться постановка дорогостоящего физического или численного эксперимента.

Многочисленные примеры такого рода дает проблема автоматизации проектирования [140—144], где вычисление одного значения функции (скалярной или векторной), описывающей проектируемую систему, требует полного расчета варианта системы. Функционирование же сложной технической системы описывается целым комплексом программ. Таким образом, при вычислении одного значения функции приходится проводить сложный и дорогостоящий счет. Достаточно точное вычисление производных функций зачастую невозможно, а может случиться, что они и не существуют. Довольно скудной, как правило, бывает и априорная информация о таких функциях.

Однако обычно, анализируя свойства реальной системы, можно обосновать предположение об ограниченности «скорости изменения» функции и охарактеризовать эту ограниченность количественно.

Специфической особенностью функций, описывающих качество работы технических объектов, является многоэкстремальность. Это свойство отражает возможность принципиально различных конструктивных решений (компоновок объекта), каждое из которых с помощью «доводки», т. е. небольшого уточнения параметров конструкции, может привести к локальному оптимуму эффективности объекта.

Приведенное неформальное описание априорной информации о функции f можно формально в весьма общем виде представить с помощью неравенства

$$|f(u)-f(v)| \leqslant \rho_0(u, v), \qquad u, v \in K, \tag{2.1}$$

где ρ_0 — некоторая фиксированная неотрицательная на $K^2 = K \times K$ функция, характеризующая ограниченность «скорости изменения» функции f.

Класс всех функций f, удовлетворяющих неравенству (2.1), будем обозначать через F_{ρ_0}. Оказывается, что рассмотрение даже такого общего класса приводит к содержательным результатам для различных задач численного анализа. Кроме того, его конкретные реализа-

ции позволяют описывать особенности многих реальных задач.

2. Представление данной априорной информации с помощью класса функций, заданных квазиметрикой. Покажем, что без ограничения общности можно считать, что функция в правой части (2.1) обладает некоторыми дополнительными свойствами, а именно свойствами квазиметрики.

Говорят, что функция ρ, определенная на K^2, обладает свойствами *квазиметрики* (является квазиметрикой) [125], если

$$\rho(u, u) = 0, \tag{2.2}$$

$$\rho(u, v) + \rho(w, v) \geqslant \rho(u, w), \tag{2.3}$$

$$\rho(u, w) = \rho(w, u), \tag{2.4}$$

$$\rho(u, v) \geqslant 0 \tag{2.5}$$

для любых $u, v, w \in K$.

Замечание. Как известно, свойства симметричности (2.4) и положительной полуопределенности (2.5) можно получить как следствие свойств (2.2) и (2.3). Действительно, из (2.3) при $u = v$ и (2.2) получаем $\rho(w, u) \geqslant \rho(u, w)$. В силу произвольности u, w справедливо также обратное неравенство, а следовательно, и равенство (2.4). Неравенство же (2.5) получается из (2.2) и (2.3) при $u = w$. $\triangle$

Теорема 2.1 [242]. *Пусть некоторая неотрицательная на K^2 функция ρ_0 задает с помощью неравенства* (2.1) *класс F_{ρ_0}. Тогда существует определенная на K^2 и обладающая свойствами квазиметрики функция ρ такая, что $F_\rho = F_{\rho_0}$, где F_ρ — класс всех функций f, удовлетворяющих неравенству*

$$|f(u) - f(v)| \leqslant \rho(u, v), \quad u, v \in K. \tag{2.6}$$

Доказательство. Определим на K^2 последовательность функций $\{\rho_i\}_{i=1}^{\infty}$ и функцию ρ следующим образом:

$$\rho_1(u, u) = 0,$$

$$\rho_1(u, v) = \min\{\rho_0(u, v), \rho_0(v, u)\} \text{ при } u \neq v,$$

$$\rho_{i+1}(u, v) = \inf_{w \in K} [\rho_i(u, w) + \rho_i(w, v)], \quad i \geqslant 1,$$

$$\rho(u, v) = \inf_{i \in \{0, 1, 2, \ldots\}} \rho_i(u, v).$$

Докажем сначала, что ρ обладает свойствами квазиметрики. По индукции легко устанавливается, что функции ρ_i при $i \geqslant 1$ обладают свойствами (2.2), (2.4), (2.5). Следовательно, теми же свойствами обладает и функция ρ.

Покажем теперь, что последовательность $\{\rho_i(u, v)\}_{i=0}^{\infty}$ монотонна при любых $u, v \in K$. Действительно, $\rho_1(u, v) \leqslant \rho_0(u, v)$ и

$$\rho_{i+1}(u, v) = \inf_{w \in K} [\rho_i(u, w) + \rho_i(w, v)] \leqslant$$
$$\leqslant \rho_i(u, u) + \rho_i(u, v) = \rho_i(u, v), \quad i \geqslant 1.$$

Отсюда следует, что для любого $\varepsilon > 0$ найдется такой номер m, что

$$\rho(u, v) + \rho(w, v) \geqslant \rho_m(u, v) + \rho_m(w, v) - \varepsilon \geqslant$$
$$\geqslant \inf_{v \in K} [\rho_m(u, v) + \rho_m(v, w)] - \varepsilon =$$
$$= \rho_{m+1}(u, w) - \varepsilon \geqslant \rho(u, w) - \varepsilon,$$

откуда, в силу произвольности $\varepsilon > 0$, получаем (2.3).

Остается установить, что $F_\rho = F_{\rho_0}$. Поскольку $\rho \leqslant \rho_0$, то из (2.6) следует (2.1), т. е. $F_\rho \subset F_{\rho_0}$. Покажем теперь справедливость обратного включения. Пусть функция f удовлетворяет неравенству (2.1). Тогда очевидно, что

$$|f(u) - f(v)| \leqslant \rho_1(u, v), \quad u, v \in K.$$

Предположив, что

$$|f(u) - f(v)| \leqslant \rho_i(u, v), \quad u, \ v \in K \tag{2.7}$$

получаем

$$|f(u) - f(v)| \leqslant |f(u) - f(w)| + |f(w) - f(v)| \leqslant$$
$$\leqslant \rho_i(u, w) + \rho_i(w, v), \quad w \in K.$$

Следовательно,

$$|f(u) - f(v)| \leqslant \inf_{w \in K} [\rho_i(u, w) + \rho_i(w, v)] = \rho_{i+1}(u, v).$$

Таким образом, неравенство (2.7) доказано по индукции для всех $i \geqslant 0$. Взяв нижнюю грань по i, получим (2.6). Итак, $F_{\rho_0} \subset F_\rho$, и потому $F_{\rho_0} = F_\rho$. $\triangle$

3. Свойства функциональных классов, заданных квазиметриками. Начнем с простого, но важного для дальнейшего факта.

Лемма 2.1. *Пусть* ρ — *произвольная квазиметрика,* $h_a(x) = -\rho(a, x)$, $a \in K$. *Тогда* $\pm h_a \in F_\rho$.

Доказательство. Воспользовавшись свойствами (2.3), (2.4), легко получим

$$|h_a(u) - h_a(v)| = |\rho(a, u) - \rho(a, v)| \leqslant \rho(u, v). \quad \triangle$$

Из доказанной леммы вытекает следующее утверждение.

Лемма 2.2. *Пусть* ρ — *произвольная квазиметрика. Тогда*

$$\sup_{f \in F_\rho} |f(u) - f(v)| = \rho(u, v).$$

Доказательство. Очевидно, что

$$\sup_{f \in F_\rho} |f(u) - f(v)| \leqslant \rho(u, v).$$

С другой стороны,

$$\sup_{f \in F_\rho} |f(u) - f(v)| \geqslant |h_u(u) - h_u(v)| = \rho(u, v). \quad \triangle$$

Следствие. *Невозможно определить класс* F_ρ *неравенством*

$$|f(u) - f(v)| \leqslant \rho'(u, v), \quad u, v \in K,$$

где $\rho'(u, v) < \rho(u, v)$ *хотя бы для одного* $(u, v) \in K^2$. $\triangle$

С помощью различных квазиметрик можно описать многие важные, с точки зрения теории и приложений, функциональные классы. Прежде чем убедиться в этом, установим следующий полезный факт.

Лемма 2.3. *Пусть функции* ρ_i *обладают свойствами квазиметрики,* $k_i \geqslant 0$, $i = 1, \ldots, s$. *Тогда свойствами квазиметрики обладают также функции* $\max\limits_{i=1, \ldots, s} (k_i \rho_i)$, $\sum\limits_{i=1}^{s} k_i \rho_i$.

Доказательство. Ограничимся проверкой свойства (2.3) для функции $\max\limits_{i=1, \ldots, s} (k_i \rho_i)$. Имеем

$$\max_{i=1, \ldots, s} (k_i \rho_i(u, v)) + \max_{i=1, \ldots, s} (k_i \rho_i(w, v)) \geqslant$$

$$\geqslant \max_{i=1, \ldots, s} (k_i \rho_i(u, v) + k_i \rho_i(w, v)) \geqslant \max_{i=1, \ldots, s} (k_i \rho_i(u, w)). \quad \triangle$$

С помощью леммы 2.3 легко проверить, что функции

$$\rho(u, v) = M \max_{t=1, \ldots, n} \{k_i |u^i - v^i|^{\alpha_i}\}, \tag{2.8}$$

$$\rho(u, v) = M \sum_{i=1}^{n} k_i |u^i - v^i|^{\alpha_i} \tag{2.9}$$

обладают свойствами квазиметрики при $M \geqslant 0$, $k_i \geqslant 0$, $0 < \alpha_i \leqslant 1$, $i = 1, \ldots, n$; $u = (u^1, \ldots, u^n)$, $v = (v^1, \ldots, v^n)$. Тем же свойством обладает, как известно, функция

$$\rho(u, v) = M\left(\sum_{i=1}^{n} (u^i - v^i)^2\right)^{1/2}. \tag{2.10}$$

С помощью квазиметрик (2.8)—(2.10) можно задать классы функций, удовлетворяющих различным вариантам условий Липшица и Гёльдера. Среди этих вариантов выделим в следующей лемме один, для которого возможность задания соответствующего класса функций с помощью квазиметрики очевидна, быть может, не сразу.

Лемма 2.4. *Класс функций, определенных на координатном параллелепипеде*

$$K = \{u = (u^1, \ldots, u^n) \mid a^i \leqslant u^i \leqslant b^i, \quad i = 1, \ldots, n\}$$

и удовлетворяющих по i-й переменной условию Липшица с константой k_i, $i = 1, \ldots, n$, совпадает с классом F_ρ, заданным квазиметрикой (2.9) *при* $M = 1$, $\alpha_1 = \alpha_2 = \ldots = \alpha_n = 1$.

Доказательство. Пусть $f \in F_\rho$, т. е.

$$|f(u) - f(v)| \leqslant \sum_{i=1}^{n} k_i |u^i - v^i|, \quad u, v \in K. \tag{2.11}$$

Тогда для любых $u^i, v^i \in [a^i, b^i]$, $w^j \in [a^j, b^j]$, $j \neq i$,

$$|f(w^1, \ldots, w^{i-1}, u^i, w^{i+1}, \ldots, w^n) - f(w^1, \ldots, w^{i-1}, v^i, w^{i+1}, \ldots, w^n)| \leqslant k_i |u^i - v^i|, \tag{2.12}$$

т. е. функция f удовлетворяет по i-й переменной условию Липшица с константой k_i, $i = 1, \ldots, n$.

Если же предположить справедливость (2.12) при $i = 1, \ldots, n$, то легко видеть, что

$$\begin{aligned}|f(u) - f(v)| \leqslant{} & |f(u^1, u^2, \ldots, u^{n-1}, u^n) - f(v^1, u^2, \ldots, u^{n-1}, u^n)| + \\ & + |f(v^1, u^2, \ldots, u^{n-1}, u^n) - f(v^1, v^2, \ldots, u^{n-1}, u^n)| + \\ & + \cdots\cdots\cdots\cdots\cdots\cdots + \\ & + |f(v^1, v^2, \ldots, v^{n-1}, u^n) - f(v^1, v^2, \ldots, v^{n-1}, v^n)| \leqslant \\ \leqslant{} & \sum_{i=1}^{n} k_i |u^i - v^i|,\end{aligned}$$

т. е. справедливо неравенство (2.11), и, следовательно, $f \in F_\rho$. △

Пусть функция $\omega(\delta)$ определена при $\delta \geqslant 0$ и обладает свойствами *модуля непрерывности*, т. е. ω непрерывна, не убывает, $\omega(0)=0$ и $\omega(\delta_1+\delta_2) \leqslant \omega(\delta_1)+\omega(\delta_2)$. В этом случае говорят, что класс функций

$$F_\omega = \{f \mid |f(u)-f(v)| \leqslant \omega(\|u-v\|),\ u,\ v \in K\} \quad (2.13)$$

задан модулем непрерывности ω (здесь $\|\cdot\|$ — некоторая норма в n-мерном координатном пространстве). Ясно, что все функции из класса F_ω равномерно непрерывны на области K, причем их модули непрерывности не превосходят ω. Рассмотрение функциональных классов, заданных модулями непрерывности, стало достаточно традиционным в теории оптимальных квадратур [7, 8, 129, 164, 191, 314].

Покажем, что эти классы являются частными случаями функциональных классов, заданных квазиметриками.

Лемма 2.5. *Пусть функция* $\omega(\delta)$ *определена при* $\delta \geqslant 0$, *не убывает*, $\omega(0)=0$, $\omega(\delta_1+\delta_2) \leqslant \omega(\delta_1)+\omega(\delta_2)$, *а функция* ρ *обладает свойствами квазиметрики. Тогда функция* $\omega(\rho)$ *также обладает свойствами квазиметрики.*

Доказательство. Имеем

$$\omega(\rho(u, v))+\omega(\rho(w, v)) \geqslant \omega(\rho(u, v)+\rho(w, v)) \geqslant \\ \geqslant \omega(\rho(u, v)),$$

и неравенство (2.3) для функции $\omega(\rho)$, таким образом, выполнено. Справедливость (2.2), (2.4), (2.5) очевидна. △

Так как функция $\rho(u, v)=\|u-v\|$, естественно, обладает свойствами квазиметрики, то из леммы 2.5 следует, что классы (2.13) являются частными случаями классов, заданных квазиметриками.

Функции $\omega(\|u-v\|)$, с помощью которых задаются классы (2.13), очевидно, инвариантны как функции переменных u, v относительно сдвига. Произвольная квазиметрика ρ таким свойством, вообще говоря, не обладает, т. е. не обязательно $\rho(u+w, v+w)=\rho(u, v)$. Это обстоятельство является существенным и позволяет задавать с помощью квазиметрик важные классы, которые нельзя задать модулями непрерывности. Примерами могут служить классы функций, удовлетворяющих в раз-

ных подобластях области определения условиям Липшица или Гёльдера с разными константами, или заданные в разных подобластях определения разными модулями непрерывности. В некоторых случаях соответствующую квазиметрику легко выписать в явном виде.

Приведем конкретный пример.

Лемма 2.6. *Пусть* $K=[0, 1]$, $0=a_0<a_1<a_2<\ldots$ $\ldots<a_s<a_{s+1}=1$, $k_1\geqslant 0, \ldots, k_{s+1}\geqslant 0$. *Тогда класс функций* f, *удовлетворяющих на отрезке* $[a_{i-1}, a_i]$ *условию Липшица с константой* k_i:

$$|f(u)-f(v)|\leqslant k_i\,|u-v|, \quad u, v\in[a_{i-1}, a_i], \quad i=1, \ldots, s+1, \tag{2.14}$$

совпадает с классом F_ρ, *заданным квазиметрикой*

$$\rho(u, v)=\rho(v, u)=$$
$$=\begin{cases} k_i\,|u-v|, \text{ если } u, v\in[a_{i-1}, a_i], \\ k_i(a_i-u)+\sum\limits_{m=i+1}^{j-1} k_m(a_m-a_{m-1})+k_j(v-a_{j-1}), \\ \text{если } u\in[a_{i-1}, a_i],\ v\in[a_{j-1}, a_j], \quad i<j. \end{cases}$$

Доказательство. Легко проверить, что функция ρ действительно обладает свойствами квазиметрики. Если $f\in F_\rho$, то неравенства (2.14), очевидно, выполняются.

Предположим теперь, что функция f удовлетворяет неравенствам (2.14), $u\in[a_{i-1}, a_i]$, $v\in[a_{j-1}, a_j]$, $i<j$. Тогда

$$|f(u)-f(v)|\leqslant|f(u)-f(a_i)|+$$
$$+\sum_{m=i+1}^{j-1}|f(a_m)-f(a_{m-1})|+|f(v)-f(a_{j-1})|\leqslant\rho(u, v),$$

то есть $f\in F_\rho$. $\triangle$

Расширяет возможности представления различных функциональных классов с помощью квазиметрик ρ и то обстоятельство, что в определении квазиметрики нет требования положительной определенности, а требуется лишь, чтобы функция ρ была неотрицательно определенной.

Пусть функциональный класс задан условиями

$$\begin{aligned} &|f(u)-f(v)|\leqslant\rho_0(u, v), \quad u, v\in K, \\ &|f(u_i)-f(v_i)|\leqslant\alpha_i, \quad \alpha_i\geqslant 0, \quad i\in I. \end{aligned} \tag{2.15}$$

Как мы знаем из теоремы 2.1, такой класс можно представить в виде F_ρ, где ρ — некоторая квазиметрика, причем если функция ρ_0 сама обладает свойствами квазиметрики, а множество I конечно, то построение ρ легко осуществляется конструктивно. С помощью второй группы условий в (2.15), полагая $\alpha_i = 0$, можно описывать равенство значений функций рассматриваемого класса в некоторых фиксированных парах точек. С помощью аналогичного приема описывается периодичность функций.

Отметим, что классы F_ρ содержат разрывные функции, если функция ρ разрывна. Так, если

$$\rho(u, v) = \begin{cases} 0, & u = v, \\ 1, & u \neq v, \end{cases}$$

то разрывная функция $h_a(x) = -\rho(a, x)$, согласно лемме 2.1, принадлежит классу F_ρ.

Как видим, возможности задания различных функциональных классов квазиметриками довольно широки. В такой общности изучение этих классов в теории численных методов было начато в [240]. В некоторых случаях оказывается целесообразным рассмотреть еще более слабые, чем (2.2)—(2.5), требования на функцию ρ. Так, в [57] ряд результатов из [240] удалось перенести на случай несимметричной функции ρ.

Отметим, что в последние годы довольно интенсивно изучаются задачи оптимизации для классов функций, удовлетворяющих разным формам условия Липшица (такие классы, как отмечалось выше, являются частными случаями функционального класса, заданного произвольной квазиметрикой). Для функций, удовлетворяющих условию Липшица, введены различные обобщения понятия градиента [329, 337], получены аналоги правила множителей Лагранжа [408], построены различные численные методы оптимизации [9, 85, 88, 89, 92, 153, 155, 199, 201, 205, 240—242, 248, 250, 265, 281, 338, 352, 427, 428], рассматриваются вопросы продолжимости и построения наилучших аппроксимаций [401].

В дальнейшем нам понадобятся следующие два утверждения.

Лемма 2.7. *Функциональный класс F_ρ, заданный произвольной квазиметрикой ρ, замкнут относительно операций взятия максимума и минимума.*

Доказательство. Пусть $f_1, f_2 \in F_\rho$. Тогда

$$|\max\{f_1(u), f_2(u)\} - \max\{f_1(v), f_2(v)\}| \leqslant$$
$$\leqslant \max\{|f_1(u) - f_1(v)|, |f_2(u) - f_2(v)|\} \leqslant \rho(u, v)$$

для любых $u, v \in K$, т. е. $\max\{f_1, f_2\} \in F_\rho$. Таким же рассуждением можно показать, что $\min\{f_1, f_2\} \in F_\rho$. △

Лемма 2.8. *Функциональный класс F_ρ, заданный произвольной квазиметрикой ρ, есть выпуклое центрально-симметричное с центром 0 множество в линейном пространстве функций, определенных на множестве K.*

Доказательство. Пусть $f_1, f_2 \in F_\rho$, $\lambda \in [0, 1]$, $u, v \in K$. Тогда

$$|\lambda f_1(u) + (1-\lambda) f_2(u) - \lambda f_1(v) - (1-\lambda) f_2(v)| \leqslant$$
$$\leqslant \lambda |f_1(u) - f_1(v)| + (1-\lambda) |f_2(u) - f_2(v)| \leqslant \rho(u, v),$$

откуда следует выпуклость F_ρ. Проверка свойства симметрии тривиальна. △

4. Другие функциональные классы. Наряду с упомянутыми мы в дальнейшем будем рассматривать и другие функциональные классы, например, класс $W^r_\infty([a, b], M)$, состоящий из функций f, $(r-1)$-я производная которых абсолютно непрерывна на $[a, b]$, а норма r-й производной в L_∞ ограничена константой M. В частности, известно, что

$$W^1_\infty([a, b], M) = \{f \mid |f(u) - f(v)| \leqslant M|u - v|, u, v \in [a, b]\}, \tag{2.16}$$

т. е. класс W^1_∞ совпадает с классом функций, удовлетворяющих на $[a, b]$ условию Липшица с константой M. Будут рассмотрены и функциональные классы совершенно иной природы.

§ 3. Классы детерминированных алгоритмов

В этом параграфе вводится общее понятие детерминированного алгоритма решения задачи приближения $S(f)$. Показано, каким образом особенности задачи, характер поступления информации о ней, а также возможности вычислителя по хранению и обработке этой информации определяют выбор класса допустимых алгоритмов решения задачи.

1. Понятие детерминированного алгоритма. Начнем с описания детерминированных алгоритмов (методов,

стратегий) α решения задачи. Работа любого алгоритма естественным образом разбивается на два этапа.

На *первом этапе* происходит сбор информации о конкретной решаемой задаче f. Для этого вычисляются значения $y_i \in Y_i$ функционалов или операторов $x_i \in X_i$, где X_i — заданное множество отображений (функционалов или операторов), определенных на F и принимающих значения из заданного множества Y_i, $i = 1, \ldots, N$. При этом N — заранее фиксированное число, определяемое ресурсами вычислителя (в дальнейшем будет рассмотрен и случай, когда N определяется в ходе вычислений).

На *втором этапе* на основе полученной информации строится приближение $\beta = \alpha(f) \in B$ к приближаемому элементу $S(f) \in B$.

Формально *детерминированный алгоритм* α определяется как совокупность отображений

$$\alpha = (\tilde{x}_1, \ldots, \tilde{x}_N, \tilde{\beta}),$$

где

$$\begin{aligned}
&\tilde{x}_1 \equiv x_1\colon F \to Y_1, \quad x_1 \in X_1,\\
&\tilde{x}_2\colon X_1 \times Y_1 \to X_2 \ni x_2\colon F \to Y_2,\\
&\tilde{x}_3\colon X_1 \times X_2 \times Y_1 \times Y_2 \to X_3 \ni x_3\colon F \to Y_3,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \qquad (3.1)\\
&\tilde{x}_N\colon X_1 \times \ldots \times X_{N-1} \times Y_1 \times \ldots \times Y_{N-1} \to X_n \ni x_N\colon F \to Y_N,\\
&\tilde{\beta}\colon X_1 \times \ldots \times X_N \times Y_1 \times \ldots \times Y_N \to B.
\end{aligned}$$

Наряду с α мы будем называть алгоритмом также отображение

$$\tilde{x}^i \stackrel{\text{def}}{=} (\tilde{x}_1, \ldots, \tilde{x}_i),$$

соответствующее первым i шагам α, $i \leqslant N$.

Решение задачи состоит в последовательном вычислении

$$\begin{aligned}
&y_1 = x_1(f),\\
&x_2 = \tilde{x}_2(x_1, y_1), \quad y_2 = x_2(f),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \qquad (3.2)\\
&x_N = \tilde{x}_N(x_1, \ldots, x_{N-1}, y_1, \ldots, y_{N-1}), \quad y_N = x_N(f),\\
&\beta = \tilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N).
\end{aligned}$$

При этом

$$\alpha(f) \stackrel{\text{def}}{=} \beta.$$

Вычисление значения $x_i(f)$ будем называть *информационным вычислением*, $i = 1, \ldots, N$. Если в явном виде не делается никаких предположений о точности информационных вычислений, то предполагается, что они выполняются без ошибки. Вычисление значения $\tilde{x}_i(x_1, \ldots, x_{i-1}, y_1, \ldots, y_{i-1})$ будем называть *алгоритмическим вычислением*, $i = 2, \ldots, N$. Вычисление значения $\tilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N)$ будем называть *итоговым вычислением* или *итоговой операцией* алгоритма [1]). Таким образом, на первом этапе работы алгоритма проводятся информационные и алгоритмические вычисления, на втором — итоговое вычисление. Проведение информационного вычисления называют еще опытом, испытанием, экспериментом, измерением и т. д. В дальнейшем мы неоднократно будем пользоваться термином «шаг алгоритма» или «шаг вычислительного процесса» применительно к совокупности алгоритмического и соответствующего ему информационного вычислений. Так, на i-м шаге алгоритма вычисляются величины $x_i = \tilde{x}_i(x_1, \ldots, x_{i-1}, y_1, \ldots, y_{i-1})$, $y_i = x_i(f)$.

2. Некоторые обозначения и термины. Введем несколько широко используемых в дальнейшем обозначений и понятий. Для $x_j \in X_j$, $y_j \in Y_j$, $j = 1, \ldots, i$, $i \leqslant N$, положим

$$x^i = (x_1, \ldots, x_i), \quad y^i = (y_1, \ldots, y_i), \quad z^i = (x^i, y^i),$$
$$F(z^i) = \{f \in F \mid x_j(f) = y_j, \quad j = 1, \ldots, i\}.$$

Вся информация вычислителя о задаче f после i шагов вычислительного процесса состоит в знании того факта, что $f \in F(z^i)$.

Будем называть вектор z^i *ситуацией после i шагов* вычислительного процесса. Ситуацию z^i назовем *реализуемой*, если $F(z^i) \neq \varnothing$. Будем говорить, что ситуация $z^i = (x^i, y^i)$ *реализуема при применении алгоритма* α (*алгоритма* $\tilde{x}^i = (\tilde{x}_1, \ldots, \tilde{x}_i)$), если ее компоненты $x_1, y_1, \ldots, x_i, y_i$ получаются в результате применения

---

[1]) Отметим, что в [275, 276] термин «алгоритм» используется лишь применительно к операции вычисления значения $\tilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N)$.

первых i формул (3.2) к некоторой функции $f \in F$, $i \leqslant N$. О векторе результатов y^i в этом случае будем также говорить, что он *реализуем при применении алгоритма* α (*алгоритма* $\tilde{x}^i$).

Отметим, что в формулах (3.1) мы для упрощения определили отображения $\tilde{x}_i$, $\tilde{\beta}$ не на минимально возможных, с точки зрения потребностей организации вычислительного процесса, множествах. Нетрудно видеть, что отображение $\tilde{x}_i$ достаточно определить на множестве ситуаций z^{i-1}, реализуемых при применении алгоритма $\tilde{x}^{i-1}$, а отображение $\tilde{\beta}$ — на множестве ситуаций z^N, реализуемых при применении алгоритма $\tilde{x}^N$. Там, где это удобно, мы будем определять эти отображения на более узких, чем в (3.1), множествах.

3. Примеры. Дадим теперь конкретные примеры фигурирующих в (3.1), (3.2) отображений x_i. Чаще всего отображению x_i соответствует вычисление тех или иных характеристик функции f в точке $x_i \in K$, например, значений функции:

$$x_i(f) = f(x_i), \tag{3.3}$$

или значений функции и ее производных до некоторого порядка:

$$x_i(f) = (f(x_i),\ f'(x_i),\ \ldots,\ f^{(r_i)}(x_i)),$$

или субдифференциала (для выпуклых функций):

$$x_i(f) = \partial f(x_i).$$

Можно рассматривать и другие случаи. Например, $x_i(f)$ есть i-й коэффициент в разложении функции f в ряд по заданной системе функции, или

$$x_i(f) = \int\limits_{K_i} f(x)\,dx,\ K_i \subset K,$$

или

$$x_i(f) = \max_{x \in K_i} f(x),\ K_i \subset K.$$

Отметим, что в последнем случае функционал x_i нелинеен. Случай нелинейной информации рассматривался в некоторых работах [276, 391], хотя рассмотрение линейной информации представляется наиболее естественным.

В соответствии с видом отображений x_i определяется множество X_i. Например, в случае (3.3) можно считать,

что

$$X_1 = X_2 = \ldots = X_N = K$$

есть множество функционалов вида (3.3), соответствующих множеству всех точек области определения K функций f из класса F.

Перейдем к описанию множеств (классов) *допустимых* алгоритмов. Выбор класса A допустимых алгоритмов решения задачи — это один из основных моментов построения модели вычислений. Этот выбор зависит от особенностей задачи, характера поступления информации, а также возможностей вычислителя по хранению и обработке этой информации.

4. Допустимые итоговые операции. Начнем с вопроса о множестве $\widehat{B}_N$ допустимых итоговых операций. Прежде всего, $\widehat{B}_N$ есть некоторое подмножество множества

$$\widetilde{B}_N \overset{\text{def}}{=} \{\widetilde{\beta} \mid \widetilde{\beta} : X_1 \times \cdots \times X_N \times Y_1 \times \cdots \times Y_n \to B\}$$

всех итоговых операций вида (3.1). Выделим две часто встречающиеся ситуации:

а) допустима единственная итоговая операция $\widetilde{\beta}$, т. е.

$$\widehat{B}_N = \{\widetilde{\beta}\}, \quad \widetilde{\beta} \in \widetilde{B}_N; \tag{3.4}$$

б) допустимы произвольные итоговые операции $\widetilde{\beta}$ вида (3.1), т. е.

$$\widehat{B}_N = \widetilde{B}_N. \tag{3.5}$$

Первая ситуация возникает, например, в задаче поиска максимума, если наряду с приближенным значением максимума необходимо указать точку, в которой максимизируемая функция f принимает это значение. В этом случае при информации $y_1 = f(x_1), \ldots, y_N = f(x_N)$ вычислитель в качестве приближения к максимальному значению выберет величину $f(x_{i_0}) = \max\limits_{i=1,\ldots,N} f(x_i)$, $1 \leqslant i_0 \leqslant N$, и укажет точку x_{i_0}. Таким образом, допустима единственная итоговая операция

$$\widetilde{\beta}(z^N) = \max_{i=1,\ldots,N} y_i. \tag{3.6}$$

Со второй ситуацией мы встретимся при рассмотрении в гл. 2, 3 задач численного интегрирования и оптимального восстановления функций.

Возможны и промежуточные ситуации, например,

$$\widetilde{B}_N = \{\widetilde{\beta} \mid \widetilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N) = \\ = p_0(x_1, \ldots, x_N) + \sum_{i=1}^{N} p_i(x_1, \ldots, x_N) y_i\}, \quad (3.7)$$

т. е. допустимые итоговые операции задаются функциями $\widetilde{\beta}$, линейными по $y^N = (y_1, \ldots, y_N)$, при этом $Y_1 = \ldots = Y_N = B = \mathbf{R}$.

5. Допустимые алгоритмические вычисления. Рассмотрим теперь вопрос о допустимых $\widetilde{x}^N = (\widetilde{x}_1, \ldots, \widetilde{x}_N)$. Здесь возможны два крайних случая.

Первый соответствует ситуации, когда при проведении любого информационного вычисления по тем или иным причинам не может быть использована информация о f, полученная при предыдущих вычислениях. Такая ситуация возникает, если, к примеру, у вычислителя нет возможности для хранения информации в оперативной памяти, или нет времени для обработки информации, или же все вычисления должны быть проведены одновременно. Последняя причина может встретиться, например, в случае, когда возможности используемой вычислительной системы позволяют одновременно проводить N вычислений или же имеется физическая или экономическая необходимость проводить все эксперименты одновременно и т. д.

В этом случае

$$\widetilde{x}_i \equiv x_i \in X_i, \quad i = 1, \ldots, N.$$

Алгоритмы (стратегии) $(x_1, \ldots, x_N)$ называют *пассивными* (*неадаптивными*). Множество всех таких алгоритмов будем обозначать через

$$X^N \stackrel{\text{def}}{=} \prod_{i=1}^{N} X_i \quad (3.8)$$

(здесь и далее символ Π используется для обозначения декартова произведения множеств). Те же термины (пассивные, неадаптивные) применяют и к алгоритмам

$$\alpha = (x_1, \ldots, x_N, \widetilde{\beta}) \in A^N \stackrel{\text{def}}{=} X^N \times \widetilde{B}_N. \quad (3.9)$$

Второй случай соответствует ситуации, когда вычислитель перед проведением любого информационного вычисления располагает информацией о результатах всех

предшествующих вычислений, достаточной памятью для хранения этих результатов и достаточным временем для их обработки, т. е. для проведения соответствующего алгоритмического вычисления.

В этом случае допустимы все алгоритмы, составленные из отображений вида (3.1). Множество всех таких алгоритмов называют *множеством всех последовательных* (*адаптивных*) алгоритмов, а его элементы — *последовательными* (*адаптивными*) алгоритмами. Будем обозначать множество всех отображений $\tilde{x}_1 \equiv x_1, \tilde{x}_2, \ldots, \tilde{x}_N$ вида (3.1) соответственно через $\tilde{X}_1 = X_1, \tilde{X}_2, \ldots, \tilde{X}_N$, а множество всех последовательных алгоритмов — через

$$\tilde{X}^N \stackrel{\text{def}}{=} \prod_{i=1}^{N} \tilde{X}_i. \tag{3.10}$$

Те же термины (последовательные, адаптивные) применяют и к алгоритмам

$$\alpha = (\tilde{x}_1, \ldots, \tilde{x}_N, \tilde{\beta}) \in \tilde{A}^N \stackrel{\text{def}}{=} \tilde{X}^N \times \tilde{B}_N. \tag{3.11}$$

Так как пассивным алгоритмам соответствуют функции-константы из $\tilde{X}^N$, то

$$X^N \subset \tilde{X}^N, \quad A^N \subset \tilde{A}^N. \tag{3.12}$$

Помимо алгоритмов из X^N и $\tilde{X}^N$ рассматриваются алгоритмы и других, «промежуточных» типов, например, алгоритмы с ограниченным объемом памяти [274, 276], алгоритмы с задержкой информации [316, 318], блочные алгоритмы [311, 317, 369, 422, 423, 463] и т. д.

Пусть информация о результатах информационных вычислений поступает с *задержкой* τ, т. е. перед проведением $(i+1)$-го информационного вычисления вычислителю становится известным результат информационного вычисления с номером $i-\tau$, $i \geqslant \tau+1$, и вычислитель имеет достаточное время для обработки поступившей информации. При проведении первых $\tau+1$ вычислений никакой информации не поступает.

В этом случае допустимы все алгоритмы $\tilde{x}^N \in \tilde{X}^N$ такие, что

$$\tilde{x}_{i+1}(z^i) = \tilde{x}_{i+1}(\bar{z}^i), \quad i \leqslant \tau, \tag{3.13}$$

для любых реализуемых ситуаций z^i, $\bar{z}^i$ и

$$\tilde{x}_{i+1}(z^i) = \tilde{x}_{i+1}(\bar{z}^i), \quad i \geqslant \tau+1, \tag{3.14}$$

для любых реализуемых ситуаций вида

$$z^i = (x_1, \ldots, x_{i-\tau}, x_{i-\tau+1}, \ldots, x_i, y_1, \ldots, y_{i-\tau}, y_{i-\tau+1}, \ldots, y_i),$$
$$\bar{z}^i = (x_1, \ldots, x_{i-\tau}, \bar{x}_{i-\tau+1}, \ldots, \bar{x}_i, y_1, \ldots, y_{i-\tau}, \bar{y}_{i-\tau+1}, \ldots, \bar{y}_i).$$

Опишем еще множество *блочных* алгоритмов. Интерес к таким алгоритмам обусловлен развитием многопроцессорных вычислительных систем и теории параллельных вычислений.

Пусть информационные вычисления распадаются на k блоков, содержащих соответственно $N_1, \ldots, N_k$ вычислений, $N_1 + \ldots + N_k = N$, причем перед началом вычислений $(j+1)$-го блока вычислителю становятся известными результаты вычислений j-го блока. Часто на практике, например для машины с r процессорами, $N_1 = \ldots = N_k = r$.

В этом случае допустимы все алгоритмы $\tilde{x}^N \in \tilde{X}^N$ такие, что

$$\tilde{x}_{i+1}(z^i) = \tilde{x}_{i+1}(\bar{z}^i) \tag{3.15}$$

для любых реализуемых ситуаций вида

$$z^i = (x_1, \ldots, x_{N_1+\ldots+N_j}, x_{N_1+\ldots+N_j+1}, \ldots, x_i, y_1, \ldots$$
$$\ldots, y_{N_1+\ldots+N_j}, y_{N_1+\ldots+N_j+1}, \ldots, y_i),$$
$$\bar{z}^i = (x_1, \ldots, x_{N_1+\ldots+N_j}, \bar{x}_{N_1+\ldots+N_j+1}, \ldots, \bar{x}_i, y_1, \ldots$$
$$\ldots, y_{N_1+\ldots+N_j}, \bar{y}_{N_1+\ldots+N_j+1}, \ldots, \bar{y}_i),$$

где

$$N_1 + \ldots + N_j < i < N_1 + \ldots + N_j + N_{j+1}.$$

§ 4. Минимаксная концепция оптимальности вычислительных алгоритмов и конкретные понятия оптимальности

Здесь мы введем различные понятия оптимальности вычислительных алгоритмов в рамках минимаксной концепции оптимальности. Будет получен ряд результатов, позволяющих прояснить структуру оптимальных алгоритмов и дать ответ на некоторые важные методологические вопросы.

1. Оптимальные по точности алгоритмы. По-видимому, первые минимаксные задачи в теории численных методов изучались П. Л. Чебышевым. Как отмечается в [190], задача построения минимаксных на классах функ-

ций квадратурных формул была поставлена А. Н. Колмогоровым. Ее изучение началось с работы [190]. Первая задача построения минимаксного последовательного алгоритма вычислений рассмотрена, по-видимому, в [371].

Рассмотрим поставленную в п. 1 § 1 задачу приближения оператора S на множестве F. Пусть $\alpha \in \widetilde{A}^N$ (см. (3.11)).

Величину

$$\sup_{f \in F} \varepsilon(\alpha, f)$$

будем называть *результатом* (точностью), *гарантированным алгоритмом* α *на классе* F, *максимальной погрешностью* α *на* F или *оценкой эффективности* α *на* F.

Пусть $\widehat{A}^N = \widehat{X}^N \times \widetilde{B}_N \subset \widetilde{A}^N = \widetilde{X}^N \times \widetilde{B}_N$ — некоторое множество алгоритмов с N информационными вычислениями (множество N-шаговых алгоритмов). Алгоритм $\alpha_0 \in \widehat{A}^N$ называется *оптимальным по точности* в $\widehat{A}^N$ на классе F, если

$$\sup_{f \in F} \varepsilon(\alpha_0, f) = \min_{\alpha \in \widehat{A}^N} \sup_{f \in F} \varepsilon(\alpha, f).$$

Пусть $\varepsilon > 0$[1]. Алгоритм $\alpha_\varepsilon \in \widehat{A}^N$ называется ε-*оптимальным по точности* в $\widehat{A}^N$ на классе F, если

$$\sup_{f \in F} \varepsilon(\alpha_\varepsilon, f) \leqslant \inf_{\alpha \in \widehat{A}^N} \sup_{f \in F} \varepsilon(\alpha, f) + \varepsilon.$$

В условиях, когда вся априорная информация вычислителя о задаче f состоит в знании того факта, что $f \in F$, минимаксная концепция оптимальности естественна и, пожалуй, наиболее оправданна с методологической точки зрения [62, 157, 235, гл. 3]. В § 6 мы обсудим высказываемые иногда утверждения о «перестраховочности» минимаксного подхода применительно к задачам организации вычислительных процессов.

2. Об учете комбинаторной сложности алгоритмов в определении оптимального по точности алгоритма. Остановимся еще на одном неформальном моменте в определении оптимальных алгоритмов. Зафиксировав число

---

[1]) Употребление здесь той же самой буквы ε, что и для обозначения критерия оценки эффективности, не приведет к каким-либо неясностям.

информационных вычислений, или, другими словами, *информационную сложность* рассматриваемых алгоритмов [186, 187, 276], мы назвали оптимальным алгоритм, гарантирующий наилучшую точность. Ниже будут рассматриваться и алгоритмы, оптимальные по числу информационных вычислений, т. е. зафиксировав необходимую точность, мы назовем оптимальным алгоритм, гарантирующий заданную точность при наименьшем числе информационных вычислений (наименьшей информационной сложности).

В обоих случаях в определении оптимальности никак не учитываются затраты ресурсов на «внутренние потребности» самих алгоритмов — стоимость алгоритмических и итогового вычислений, или, другими словами, *комбинаторная сложность* рассматриваемых алгоритмов [276].

На первый взгляд это недопустимо. Однако в вычислительной практике, как уже отмечалось, встречается множество сложных задач, для которых проведение одного информационного вычисления требует многих секунд и даже минут машинного времени. В таких задачах пренебрежение затратами ресурсов на внутренние потребности алгоритмов часто оказывается оправданным.

Даже если это не так, и такими затратами пренебречь нельзя, все равно отыскание оптимального в смысле данного выше определения и последовательно-оптимального (определение будет дано в § 6) методов решения задачи является весьма полезным, так как позволяет выявить предельные возможности вычислителя. На этой основе часто удается построить алгоритмы, в том или ином смысле близкие к оптимальным и в то же время обладающие приемлемой комбинаторной сложностью. Их применение на практике дает во многих случаях хорошие результаты. Примеры построения такого рода алгоритмов встречаются во всех последующих главах.

Конечно, в идеале при выборе концепции оптимальности следовало бы учитывать как информационную, так и комбинаторную сложность алгоритмов. Однако до сих пор достаточно содержательных результатов в этом направлении немного.

Хуже того, будучи до конца последовательными, на этом пути мы едва ли сможем построить замкнутую модель вычислений. Действительно, учитывая в модели

комбинаторную сложность алгоритмов, т. е. стоимость вычисления $\tilde{x}_i(z^{i-1})$, $i=2, \ldots, N$, $\tilde{\beta}(z^N)$, необходимо ставить вопрос об оптимальном методе вычисления этих величин. Например, чтобы найти $\tilde{x}_i(z^{i-1})$, в некоторых случаях, как мы увидим ниже, приходится решать задачу целочисленной оптимизации. Значит, встает вопрос об оптимальном методе решения таких задач и т. д. В конце концов мы придем к трудным самостоятельным проблемам оптимального вычисления значений элементарных функций, оптимального выполнения арифметических операций и др., а также необходимости учета специфики конкретной ЭВМ. Кроме того, совсем уж трудно формализовать вопрос о стоимости исследовательских усилий по построению оптимальных алгоритмов.

3. О вероятностных концепциях оптимальности. Помимо минимаксной, сформулированы и подвергаются изучению и другие концепции оптимальности. Например, если алгоритм предназначен для решения серии однотипных задач, причем имеются априорные представления о частотах появления этих задач, то можно пытаться задать априорное вероятностное распределение на классе задач и строить оптимальные в среднем по тому или иному критерию алгоритмы. Такой подход, называемый вероятностным, статистическим, информационно-статистическим, байесовским и т. п., развивается, например, в работах [100—102, 184, 238, 265, 297, 306—308, 319, 330, 342, 361, 376, 377, 399, 400, 450, 455, 456, 467, 468]. Большинство из перечисленных работ посвящено задаче поиска экстремума функций одной и нескольких переменных. В настоящее время правомерность использования статистических моделей многоэкстремальных целевых функций в задачах глобальной оптимизации обоснована аксиоматически на основе априорной информации о целевых функциях, не включающей в себя информацию о вероятностном распределении на классе рассматриваемых целевых функций [102].

4. Одно вспомогательное утверждение. В дальнейшем нам понадобится следующая почти очевидная лемма, которая близка к соответствующему утверждению из [62]. Пусть X, Y — множества произвольной природы, $g\colon X \times Y \to \mathbf{R}$, $\Phi = \{\varphi | \varphi\colon Y \to X\}$ — множество всех отображений из Y в X. Определим функцию g на множестве $\Phi \times Y$ формулой $g(\varphi, y) = g(\varphi(y), y)$.

Лемма 4.1. *Справедливо равенство*

$$\inf_{\varphi\in\Phi}\sup_{y\in Y} g(\varphi, y) = \sup_{y\in Y}\inf_{x\in X} g(x, y).$$

Если при этом существует функция $\varphi_0 \in \Phi$ *такая, что*

$$g(\varphi_0(y), y) = \inf_{x\in X} g(x, y), \quad y \in Y,$$

то

$$\sup_{y\in Y} g(\varphi_0, y) = \inf_{\varphi\in\Phi}\sup_{y\in Y} g(\varphi, y).$$

Доказательство. Для произвольных $\varphi \in \Phi$ и $y \in Y$

$$g(\varphi, y) = g(\varphi(y), y) \geqslant \inf_{x\in X} g(x, y),$$

откуда

$$\sup_{y\in Y} g(\varphi, y) \geqslant \sup_{y\in Y}\inf_{x\in X} g(x, y)$$

и

$$\inf_{\varphi\in\Phi}\sup_{y\in Y} g(\varphi, y) \geqslant \sup_{y\in Y}\inf_{x\in X} g(x, y). \tag{4.1}$$

Определим теперь для произвольного $\varepsilon > 0$ функцию φ_ε: $Y \to X$ так, чтобы для любого $y \in Y$ выполнялись неравенства

$$g(\varphi_\varepsilon, y) = g(\varphi_\varepsilon(y), y) \leqslant \inf_{x\in X} g(x, y) + \varepsilon,$$

$$\text{если } \inf_{x\in X} g(x, y) > -\infty,$$

$$g(\varphi_\varepsilon, y) = g(\varphi_\varepsilon(y), y) \leqslant -1/\varepsilon, \text{ если } \inf_{x\in X} g(x, y) = -\infty.$$

В силу произвольности $y \in Y$

$$\sup_{y\in Y} g(\varphi_\varepsilon, y) \leqslant \max\left\{\sup_{y\in Y}\inf_{x\in X} g(x, y) + \varepsilon, -1/\varepsilon\right\}$$

и, следовательно,

$$\inf_{\varphi\in\Phi}\sup_{y\in Y} g(\varphi, y) \leqslant \max\left\{\sup_{y\in Y}\inf_{x\in X} g(x, y) + \varepsilon, -1/\varepsilon\right\}.$$

Устремляя ε к нулю, получаем

$$\inf_{\varphi\in\Phi}\sup_{y\in Y} g(\varphi, y) \leqslant \sup_{y\in Y}\inf_{x\in X} g(x, y). \tag{4.2}$$

Неравенства (4.1) и (4.2) доказывают первое утверждение

леммы. В силу определения φ_0 имеем

$$\sup_{y\in Y} g(\varphi_0(y), y) = \sup_{y\in Y}\inf_{x\in X} g(x, y).$$

Отсюда и из первого утверждения леммы следует второе утверждение. Δ

5. Центральная итоговая операция. В п. 1 мы назвали алгоритм $\alpha_0 = (\tilde{x}_0^N, \tilde{\beta}_0) \in \widehat{A}^N = \widehat{X}^N \times \widehat{B}_N$, реализующий минимум по $\tilde{x}^N$, $\tilde{\beta}$ в выражении

$$\min_{\tilde{x}^N\in\widehat{X}^N, \tilde{\beta}\in\widehat{B}_N} \sup_{f\in F} \varepsilon((\tilde{x}^N, \tilde{\beta}), f),$$

оптимальным по точности в $\widehat{A}^N$ на классе F. Как нетрудно понять, из оптимальности $(\tilde{x}_0^N, \tilde{\beta}_0)$, вообще говоря, не следует, что $\tilde{\beta}_0$ реализует минимум в выражении $\min\limits_{\tilde{\beta}\in\widehat{B}_N} \sup\limits_{f\in F} \varepsilon((\tilde{x}^N, \tilde{\beta}), f)$ при $\tilde{x}^N \neq \tilde{x}_0^N$, даже если при некотором $\tilde{\beta}$ алгоритм $(\tilde{x}^N, \tilde{\beta})$ оптимален по точности в $\widehat{A}^N$. В указанном смысле $\tilde{\beta}_0$ не является универсально оптимальной итоговой операцией.

Пусть $\widehat{B}_N = \tilde{B}_N$, т. е. произвольные итоговые операции допустимы. В этом случае универсально оптимальная (либо при любом $\varepsilon > 0$ — универсально ε-оптимальная) итоговая операция существует (точный смысл термина — в теореме 4.1). Построим ее.

Пусть информационные вычисления завершены и в результате их проведения сложилась ситуация z^N. Следовательно, вся информация вычислителя о задаче f после завершения информационных вычислений состоит в знании того факта, что $f \in F(z^N)$. В этом случае очевидно, что наилучшая гарантированная точность есть

$$\varepsilon_N(z^N) = \inf_{\beta\in B} \sup_{f\in F(z^N)} \gamma(S(f), \beta). \tag{4.3}$$

Предположим, что при любом реализуемом z^N нижняя грань в (4.3) достигается. Элемент множества B, на котором она достигается, будем обозначать через $\tilde{\beta}_*(z^N)$ (если таких элементов несколько, то фиксируется и обозначается через $\tilde{\beta}_*(z^N)$ любой из них). Таким образом, при любом реализуемом z^N

$$\sup_{f\in F(z^N)} \gamma(S(f), \beta_*(z^N)) = \inf_{\beta\in B} \sup_{f\in F(z^N)} \gamma(S(f), \beta). \tag{4.4}$$

Итоговую операцию $\widetilde{\beta}_*$ будем называть *центральной итоговой операцией*[1]).

Аналогом понятия центральной итоговой операции в случае, когда нижняя грань в (4.3) не достигается, служит понятие *ε-центральной итоговой операции* $\widetilde{\beta}_*^{\varepsilon}$, для которой

$$\sup_{f\in F(z^N)} \gamma\left(S(f), \widetilde{\beta}_*^{\varepsilon}(z^N)\right) \leqslant \inf_{\beta\in B} \sup_{f\in F(z^N)} \gamma(S(f), \beta) + \varepsilon$$

при любом реализуемом z^N.

Чтобы пояснить употребление термина «центральная итоговая операция», напомним, что величина $r = \inf\limits_{\beta\in B} \sup\limits_{s\in S} \gamma(s, \beta)$ называется *чебышевским радиусом* множества S, лежащего в метрическом пространстве B с метрикой γ. Если существует точка β_* такая, что $\sup\limits_{s\in S} \gamma(s, \beta_*) = r$, то она называется *чебышевским центром* множества S. Грубо говоря, r — радиус наименьшего шара, содержащего S, а β_* — центр этого шара. Отметим, что точка β_* может определяться не единственным образом.

Образ $S(F(z^N))$ множества $F(z^N)$ при отображении S является для вычислителя по окончании информационных вычислений «множеством неопределенности»: все его элементы (и только элементы этого множества) могут оказаться (при информации $f\in F(z^N)$, имеющейся у вычислителя) приближаемым элементом $S(f)$. Переписывая (4.3) в виде

$$\varepsilon_N(z^N) = \inf_{\beta\in B} \sup_{s\in S(F(z^N))} \gamma(s, \beta),$$

убеждаемся, что величина $\varepsilon_N(z^N)$ является чебышевским радиусом множества $S(F(z^N))$, а точка $\widetilde{\beta}_*(z^N)$ — его чебышевским центром. Именно поэтому $\widetilde{\beta}_*$ называется центральной итоговой операцией.

Следующая теорема устанавливает свойство универсальной оптимальности центральной итоговой операции.

---

[1]) Одно и то же обозначение $\widetilde{\beta}_*$ для центральной итоговой операции в дальнейшем будет использоваться при различных N. Это не приведет к недоразумениям, хотя индекс N в принятом обозначении и отсутствует. Каждый раз из контекста будет ясно, сколько аргументов у функции $\widetilde{\beta}_*$.

Теорема 4.1. *При любом* $\tilde{x}^N \in \tilde{X}^N$

$$\min_{\tilde{\beta} \in \tilde{B}_N} \sup_{f \in F} \varepsilon\left((\tilde{x}^N, \tilde{\beta}), f\right) = \sup_{f \in F} \varepsilon\left((\tilde{x}^N, \tilde{\beta}_*), f\right),$$

где $\tilde{\beta}_*$ — *центральная итоговая операция.*

Доказательство. Зафиксируем произвольное $\tilde{x}^N \in \tilde{X}^N$. Для доказательства теоремы достаточно обосновать следующую цепочку равенств:

$$\inf_{\tilde{\beta} \in \tilde{B}_N} \sup_{f \in F} \varepsilon\left((\tilde{x}^N, \tilde{\beta}), f\right) = \inf_{\tilde{\beta} \in \tilde{B}_N} \sup_{f \in F} \gamma\left(S(f), \tilde{\beta}(z^N)\right) =$$

$$= \inf_{\tilde{\beta} \in \tilde{B}_N} \sup_{z^N \in \{z^N\}_{\tilde{x}^N}} \sup_{f \in F(z^N)} \gamma\left(S(f), \tilde{\beta}(z^N)\right) =$$

$$= \sup_{z^N \in \{z^N\}_{\tilde{x}^N}} \inf_{\beta \in B} \sup_{f \in F(z^N)} \gamma(S(f), \beta) =$$

$$= \sup_{z^N \in \{z^N\}_{\tilde{x}^N}} \sup_{f \in F(z^N)} \gamma\left(S(f), \tilde{\beta}_*(z^N)\right) =$$

$$= \sup_{f \in F} \gamma\left(S(f), \tilde{\beta}_*(z^N)\right) = \sup_{f \in F} \varepsilon\left((\tilde{x}^N, \tilde{\beta}_*), f\right).$$

Через $z^N = (x_1, \ldots, x_N, y_1, \ldots, y_N)$ в правой части первого равенства обозначен вектор, который получается по формулам (3.2) при заданных $\tilde{x}^N$ и f, т. е. в этом равенстве используется лишь определение (1.1) критерия ε с учетом определения $\alpha(f)$ по формулам (3.2).

Через $\{z^N\}_{\tilde{x}^N}$ мы обозначаем здесь множество всех z^N, реализуемых при применении алгоритма $\tilde{x}^N$ (см. п. 2 § 3). Следовательно, второе равенство обосновывается с помощью очевидного соотношения

$$F = \bigcup_{z^N \in \{z^N\}_{\tilde{x}^N}} F(z^N). \tag{4.5}$$

Справедливость третьего равенства устанавливается леммой 4.1, примененной к функции $g(\beta, z^N) = \sup_{f \in F(z^N)} \gamma(S(f), \beta)$. Четвертое равенство следует из (4.4); пятое — из (4.5), при этом смысл вектора z^N в правой его части тот же, что и в правой части первого равенства. Наконец, в шестом равенстве, как и в первом, используется лишь определение критерия ε. △

Следствие. *Если алгоритм* $(\tilde{x}_0^N, \tilde{\beta}_0)$ *оптимален по точности в* $\tilde{X}^N \times \tilde{B}_N$ *на классе* F, $\tilde{\beta}_*$ — *центральная итоговая операция, то алгоритм* $(\tilde{x}_0^N, \tilde{\beta}_*)$ *также оптимален по точности в* $\tilde{X}^N \times \tilde{B}_N$ *на классе* F.

Доказательство. В цепочке равенств

$$\inf_{\tilde{x}^N \in \tilde{X}^N} \inf_{\tilde{\beta} \in \tilde{B}_N} \sup_{f \in F} \varepsilon((\tilde{x}^N, \tilde{\beta}), f) = \sup_{f \in F} \varepsilon((\tilde{x}_0^N, \tilde{\beta}_0), f) =$$
$$= \inf_{\tilde{\beta} \in \tilde{B}_N} \sup_{f \in F} \varepsilon((\tilde{x}_0^N, \tilde{\beta}), f) = \sup_{f \in F} \varepsilon((\tilde{x}_0^N, \tilde{\beta}_*), f)$$

первое и второе равенства выражают оптимальность алгоритма $(\tilde{x}_0^N, \tilde{\beta}_0)$, справедливость третьего следует из теоремы 4.1, равенство же между крайними слева и справа в цепочке выражениями доказывает утверждение следствия. △

6. Оптимальность линейной итоговой операции. Пусть S — функционал, $\gamma(S(f), \beta) = |S(f) - \beta|$. Предположим, что используется пассивный алгоритм $x^N = (x_1, \ldots, x_N) \in X^N$, где $x_1, \ldots, x_N$ — фиксированные функционалы. Тогда итоговую операцию можно рассматривать как функцию θ переменных

$$y_1 = x_1(f), \ldots, y_N = x_N(f): \ \theta(y_1, \ldots, y_N) = \tilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N).$$

Оптимальной итоговой операцией при фиксированных $x_1, \ldots, x_N$ назовем такую итоговую операцию θ_0, что

$$\sup_{f \in F} |S(f) - \theta_0(x_1(f), \ldots, x_N(f))| = \min_{\theta \in \Theta} \sup_{f \in F} |S(f) - \theta(x_1(f), \ldots, x_N(f))|, \quad (4.6)$$

где Θ — множество всех числовых функций θ переменных

$$(y_1, \ldots, y_N) \in \{(y_1, \ldots, y_N) \mid y_1 = x_1(f), \ldots, y_N = x_N(f), \ f \in F\}.$$

В силу теоремы 4.1 центральная итоговая операция $\tilde{\beta}_*(x_1, \ldots, x_N, y_1, \ldots, y_N)$, рассматриваемая как функция переменных $y_1, \ldots, y_N$ при фиксированных $x_1, \ldots, x_N$, оптимальна в смысле данного определения. Однако оптимальная итоговая операция при фиксированных $x_1, \ldots, x_N$ может быть не единственной. Поэтому возни-

кает вопрос о выборе среди оптимальных итоговых операций функции θ_0 по возможности более простого вида. Такой функцией является, в первую очередь, линейная функция. Оказывается, что для одного важного класса задач оптимальная линейная итоговая операция существует. Соответствующее утверждение формулируется в следующей теореме, являющейся обобщением аналогичного результата из [228] (см. также [20]).

Теорема 4.2 [445]. *Пусть F — выпуклое множество в линейном пространстве, функционалы S, x_1, ..., x_N линейны. Тогда среди оптимальных итоговых операций при фиксированных x_1, ..., x_N найдется линейная функция $\theta_0(y_1, \ldots, y_N) = p_0 + p_1 y_1 + \ldots + p_N y_N$. При этом если множество F центрально-симметрично с центром 0, то $p_0 = 0$.*

Доказательство. Положим

$$Y = \{y = (y_0, y_1, \ldots, y_N) \mid y_0 = S(f), \\ y_1 = x_1(f), \ldots, y_N = x_N(f), \ f \in F\}.$$

Выпуклость множества F и линейность функционалов S, x_1, ..., x_N влекут за собой выпуклость множества $Y \subset \mathbf{R}^{N+1}$. Для произвольной функции $\theta \in \Theta$ имеем

$$\sup_{f \in F} |S(f) - \theta(x_1(f), \ldots, x_N(f))| = \\ = \sup_{(y_0, y_1, \ldots, y_N) \in Y} |y_0 - \theta(y_1, \ldots, y_N)| = \\ = \sup_{(y_1, \ldots, y_N) \in \pi(Y)} \ \sup_{y_0 \in \sigma(y_1, \ldots, y_N)} |y_0 - \theta(y_1, \ldots, y_N)|,$$

где $\sigma(y_1, \ldots, y_N) = \{y_0 \mid (y_0, y_1, \ldots, y_N) \in Y\}$ — промежуток с концами

$$a(y_1, \ldots, y_N) = \inf_{(y_0, y_1, \ldots, y_N) \in Y} y_0,$$

$$b(y_1, \ldots, y_N) = \sup_{(y_0, y_1, \ldots, y_N) \in Y} y_0,$$

а $\pi(Y) = \{(y_1, \ldots, y_N) \mid \sigma(y_1, \ldots, y_N) \neq \varnothing\}$ — проекция множества Y на подпространство переменных y_1, ..., y_N. Следовательно, с учетом леммы 4.1,

$$\inf_{\theta \in \Theta} \sup_{f \in F} |S(f) - \theta(x_1(f), \ldots, x_N(f))| = \\ = \inf_{\theta \in \Theta} \ \sup_{(y_1, \ldots, y_N) \in \pi(Y)} \ \sup_{y_0 \in \sigma(y_1, \ldots, y_N)} |y_0 - \theta(y_1, \ldots, y_N)| =$$

$$= \sup_{(y_1,\ldots,y_N)\in\pi(Y)} \inf_{r\in\mathbf{R}} \sup_{y_0\in\sigma(y_1,\ldots,y_N)} |y_0 - r| =$$

$$= \sup_{(y_1,\ldots,y_N)\in\pi(Y)} \inf_{r\in\mathbf{R}} \max\{b(y_1,\ldots,y_N)-r,\, r-a(y_1,\ldots,y_N)\} =$$

$$= \sup_{(y_1,\ldots,y_N)\in\pi(Y)} \frac{1}{2}(b(y_1,\ldots,y_N) - a(y_1,\ldots,y_N)) \stackrel{\text{def}}{=} d. \quad (4.7)$$

Если $d=+\infty$, то любая функция $\theta\in\Theta$ оптимальна. Пусть $d<+\infty$. В этом случае, для того чтобы убедиться в существовании линейной оптимальной итоговой операции, достаточно указать функцию $\theta_0(y_1, \ldots, y_N)=p_0+ + p_1y_1+\ldots+p_Ny_N$ такую, что

$$\sup_{f\in F} |S(f) - \theta_0(x_1(f), \ldots, x_N(f))| \leqslant d$$

или, иными словами,

$$|y_0-(p_0+p_1y_1+\ldots+p_Ny_N)| \leqslant d \quad \text{при всех}$$

$$(y_0, y_1, \ldots, y_N)\in Y. \quad (4.8)$$

Рассмотрим множества $Y_1=Y-(d, 0, \ldots, 0)$, $Y_2= =Y+(d, 0, \ldots, 0)$ (рис. 1). Пусть $y=(y_0, y_1, \ldots, y_N)\in$

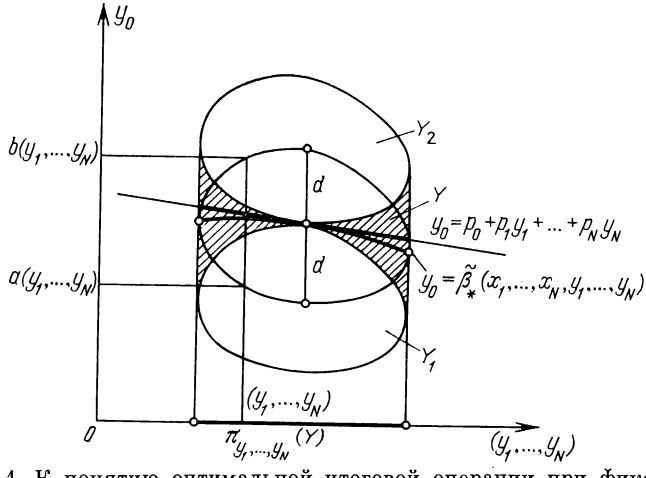

Рис. 1. К понятию оптимальной итоговой операции при фиксированных $x_1, \ldots, x_N$

$\in \operatorname{ri} Y_1 \cap \operatorname{ri} Y_2$ (как обычно, $\operatorname{ri} A$ обозначает относительную внутренность множества A). Тогда $(y_0+d, y_1, \ldots \ldots, y_N)\in \operatorname{ri} Y$, $(y_0-d, y_1, \ldots, y_N)\in \operatorname{ri} Y$. Прямая, соединяющая данные точки, очевидно, принадлежит аффин-

ной оболочке множества Y. На этой же прямой лежат и точки $(y_0+d+\varepsilon/2,\ y_1,\ \ldots,\ y_N)$, $(y_0-d-\varepsilon/2,\ y_1,\ \ldots,\ y_N)$. Следовательно, при достаточно малом $\varepsilon>0$ они являются элементами множества Y, откуда

$$b(y_1,\ \ldots,\ y_N)-a(y_1,\ \ldots,\ y_N)\geqslant(y_0+d+\varepsilon/2)-$$
$$-(y_0-d-\varepsilon/2)=2d+\varepsilon,$$

что противоречит последнему равенству в (4.7). Итак, $\operatorname{ri} Y_1\cap\operatorname{ri} Y_2=\varnothing$.

По теореме отделимости (см., например, [255]) в этом случае существует гиперплоскость $\langle c,\ y\rangle=\xi$, $c=(c_0,\ c_1,\ \ldots,\ c_N)$, разделяющая множества Y_1 и Y_2 в собственном смысле, т. е.

$$\langle c,\ y'\rangle\leqslant\xi\leqslant\langle c,\ y''\rangle \quad \text{при всех} \quad y'\in Y_1,\quad y''\in Y_2, \tag{4.9}$$

$$\langle c,\ \bar{y}'\rangle<\langle c,\ \bar{y}''\rangle \quad \text{при некоторых} \ \bar{y}'\in Y_1,\ \bar{y}''\in Y_2. \tag{4.10}$$

Пусть $c_0=0$. Возьмем произвольный вектор $y'=(y_0',y_1',\ldots,y_N')\in Y_1$ и $y''=(y_0'+2d,y_1',\ldots,y_N')\in Y_2$. Имеем $\langle c,\ y'\rangle=\langle c,\ y''\rangle$. Следовательно, в силу (4.9) $\langle c,\ y'\rangle=\xi$ при всех $y'\in Y_1$. Аналогично, $\langle c,\ y''\rangle=\xi$ при всех $y''\in Y_2$, а это противоречит (4.10). Полученное противоречие показывает, что $c_0\neq 0$.

Пусть для определенности $c_0>0$. Полагая $p_1=-c_1/c_0,\ \ldots,\ p_N=-c_N/c_0$, $p_0=\xi/c_0$ и используя определение множеств Y_1, Y_2, перепишем (4.9) в виде

$$y_0-d-\sum_{i=1}^{N}p_iy_i\leqslant p_0\leqslant y_0+d-\sum_{i=1}^{N}p_iy_i$$
$$\text{при всех } y=(y_0,y_1,\ldots,y_N)\in Y,$$

что показывает справедливость (4.8).

Для завершения доказательства осталось убедиться, что $p_0=0$ в случае центрально-симметричного множества F с центром 0. В этом случае множество Y также центрально-симметрично с центром 0. Пусть $(y_1,\ \ldots,\ y_N)\in\pi(Y)$. Тогда точки $(a(y_1,\ \ldots,\ y_N),\ y_1,\ \ldots,\ y_N)$, $(b(y_1,\ \ldots,\ y_N),\ y_1,\ \ldots,\ y_N)$ принадлежат замыканию $\bar{Y}$ множества Y. Нетрудно видеть, что множество $\bar{Y}$ также центрально-симметрично с центром 0. Следовательно, $(-a(y_1,\ \ldots,\ y_N),\ -y_1,\ \ldots,\ -y_N)\in\bar{Y}$. Как известно, вместе с множеством Y выпукло и множество $\bar{Y}$ (см., например,

[255]), и потому

$$\frac{1}{2}(b(y_1, \ldots, y_N), y_1, \ldots, y_N) + \frac{1}{2}(-a(y_1, \ldots, y_N), -y_1, \ldots$$

$$\ldots, -y_N) = \left(\frac{b(y_1, \ldots, y_N) - a(y_1, \ldots, y_N)}{2}, 0, \ldots, 0\right) \in \overline{Y},$$

а в силу симметрии $\overline{Y}$ и

$$\left(\frac{a(y_1, \ldots, y_N) - b(y_1, \ldots, y_N)}{2}, 0, \ldots, 0\right) \in \overline{Y}.$$

По определению функций a и b

$$b(0, \ldots, 0) - a(0, \ldots, 0) \geqslant$$

$$\geqslant \frac{b(y_1, \ldots, y_N) - a(y_1, \ldots, y_N)}{2} - \frac{a(y_1, \ldots, y_N) - b(y_1, \ldots, y_N)}{2} =$$

$$= b(y_1, \ldots, y_N) - a(y_1, \ldots, y_N),$$

откуда, в силу данного в (4.7) определения величины d, произвольности $(y_1, \ldots, y_N) \in \pi(Y)$ и симметрии множества Y, получаем последовательно, что $b(0, \ldots, 0) - a(0, \ldots, 0) = 2d$, $b(0, \ldots, 0) = -a(0, \ldots, 0) = d$.

При $y_1 = \ldots = y_N = 0$ неравенство (4.8) принимает вид $|y_0 - p_0| \leqslant d$ при всех $y_0 \in (a(0, \ldots, 0), b(0, \ldots, 0)) = (-d, d)$, что возможно лишь при $p_0 = 0$. △

На рис. 1 жирными линиями выделены графики функции $y_0 = \widetilde{\beta}_*(x_1, \ldots, x_N, y_1, \ldots, y_N)$, которая задает центральную итоговую операцию ($x_1, \ldots, x_N$ фиксированы), а также функции $y_0 = p_0 + p_1 y_1 + \ldots + p_N y_N$, которая задает линейную оптимальную итоговую операцию при фиксированных $x_1, \ldots, x_N$. Очевидно, что любая функция $y_0 = \theta_0(y_1, \ldots, y_N)$, график которой целиком лежит в заштрихованном на рис. 1 множестве, задает оптимальную итоговую операцию при фиксированных $x_1, \ldots, x_N$, поскольку $|y_0 - \theta_0(y_1, \ldots, y_N)| \leqslant d$ при всех $(y_0, y_1, \ldots, y_N) \in Y$.

7. Замечание к понятию оптимальной итоговой операции. Данное в предыдущем пункте определение (4.6) оптимальной итоговой операции при фиксированных $x_1, \ldots, x_N$ очевидным образом распространяется на случай произвольных $x^N = (x_1, \ldots, x_N) \in X^N$. Именно, итоговую операцию $\widetilde{\beta}_{\text{opt}}$ такую, что при всех $x^N \in X^N$

$$\sup_{f \in F} \gamma(S(f), \widetilde{\beta}_{\text{opt}}(x_1, \ldots, x_N, x_1(f), \ldots, x_N(f))) =$$

$$= \min_{\widetilde{\beta} \in \widetilde{B}_N} \sup_{f \in F} \gamma(S(f), \widetilde{\beta}(x_1, \ldots, x_N, x_1(f), \ldots, x_N(f))), \quad (4.11)$$

можно назвать *итоговой операцией*, *оптимальной в множестве* $\widetilde{B}_N$.

Однако данное определение целесообразно использовать лишь в случае, когда применяются только пассивные алгоритмы $(x^N, \widetilde{\beta}) \in A^N$ (см. (3.9)). Дело в том, что определенная указанным способом оптимальная итоговая операция не обладает свойством универсальной оптимальности, которым, в соответствии с п. 5, обладает центральная итоговая операция. Именно, как показывает следующий пример, из оптимальности некоторого алгоритма $(\widetilde{x}_0^N, \widetilde{\beta}_0)$ не вытекает, вообще говоря, оптимальность алгоритма $(\widetilde{x}_0^N, \widetilde{\beta}_{\text{opt}})$ (ср. со следствием теоремы 4.1).

Пример. Пусть $F = \{0, 2, 4, 6, 8, 10, 12, 14\}$, $S(f) = f$, $B = \mathbf{R}$ (напомним, что оператор S принимает значения в метрическом пространстве B с метрикой γ), $N = 2$, $X_1 = X_2 = \{a_1, \ldots, a_7\}$, где

$$a_i(f) = \begin{cases} 0, & f < 2i - 1, \\ 1, & f > 2i - 1, \end{cases} \quad i = 1, \ldots, 7.$$

Положим $\widetilde{\beta}_{\text{opt}}(a_4, a_6, 1, 0) = \widetilde{\beta}_{\text{opt}}(a_6, a_4, 0, 1) = 10$. При всех остальных $z^2 = (x_1, x_2, y_1, y_2)$ положим $\widetilde{\beta}_{\text{opt}}(x_1, x_2, y_1, y_2) = \widetilde{\beta}_*(x_1, x_2, y_1, y_2)$, где $\widetilde{\beta}_*$ — центральная итоговая операция. Нетрудно проверить, что $\widetilde{\beta}_{\text{opt}}$ является оптимальной итоговой операцией в смысле определения (4.11). В частности,

$$\sup_{f \in F} \gamma(S(f), \widetilde{\beta}_{\text{opt}}(a_4, a_6, a_4(f), a_6(f))) =$$
$$= \min_{\widetilde{\beta} \in \widetilde{B}_N} \sup_{f \in F} \gamma(S(f), \widetilde{\beta}(a_4, a_6, a_4(f), a_6(f))) = 3,$$

причем верхняя грань в левой части последнего равенства достигается при $f = 0$ и при $f = 6$.

Определим теперь последовательный алгоритм $(\widetilde{x}_1^0, \widetilde{x}_2^0)$ следующим образом: $\widetilde{x}_1^0 = a_4$, $\widetilde{x}_2^0(a_4, 0) = a_2$, $\widetilde{x}_2^0(a_4, 1) = a_6$. Ясно, что алгоритм $(\widetilde{x}_1^0, \widetilde{x}_2^0, \widetilde{\beta}_*)$ гарантирует точность 1 (см. п. 1) и оптимален по точности в $\widetilde{A}^2$ (см. (3.11)) на классе F. В то же время алгоритм $(\widetilde{x}_1^0, \widetilde{x}_2^0, \widetilde{\beta}_{\text{opt}})$ гарантирует худшую точность (максимальная погрешность этого алгоритма равна 2, причем максимум достигается при $f = 8$) и, таким образом, не оптимален. $\triangle$

В выражении для наилучшего гарантированного алгоритмами из $\widehat{A}^N$ результата (точности)

$$\inf_{\alpha\in\widehat{A}^N}\sup_{f\in F}\varepsilon(\alpha, f) \tag{4.12}$$

фигурирует операция взятия нижней грани по множеству $\widehat{A}^N$. Оно во многих случаях, например, когда $\widehat{A}^N = \widetilde{A}^N$ (см. (3.11)) или когда $\widehat{A}^N$ задается условиями (3.13) и (3.14) или же (3.15), является множеством сложной функциональной структуры.

Зададимся целью представить величину (4.12) в виде кратного минимакса, точнее, в виде последовательности операций вычисления нижних и верхних граней критерия ε по множествам, структура которых значительно проще структуры множества $\widehat{A}^N$. Решение этой задачи, содержащееся в п.п. 8—11, основано на лемме 4.1. Оно позволит прояснить в дальнейшем некоторые важные, с точки зрения теории, вопросы.

8. Наилучшая гарантированная точность в классе пассивных алгоритмов. Рассмотрим класс пассивных алгоритмов A^N (см. (3.9)), считая произвольные итоговые операции $\widetilde{\beta}\in\widetilde{B}_N$ допустимыми. Напомним, что величина $\varepsilon_N(z^N)$ определена равенством (4.3).

Теорема 4.3. *Справедливо равенство*

$$\inf_{\alpha\in A^N}\sup_{f\in F}\varepsilon(\alpha, f) = \inf_{x^N\in X^N}\sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}}\varepsilon_N(z^N) =$$
$$= \inf_{x^N\in X^N}\sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}}\inf_{\beta\in B}\sup_{f\in F(z^N)}\gamma(S(f), \beta).$$

Доказательство. Очевидно, что

$$\inf_{\alpha\in A^N}\sup_{f\in F}\varepsilon(\alpha, f) =$$
$$= \inf_{x^N\in X^N}\inf_{\widetilde{\beta}\in\widetilde{B}_N}\sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}}\sup_{f\in F(z^N)}\gamma(S(f), \widetilde{\beta}(z^N)).$$

Зафиксируем теперь $x^N\in X^N$ и применим лемму 4.1 к функции

$$g(\widetilde{\beta}, y^N) = \sup_{f\in F(z^N)}\gamma(S(f), \widetilde{\beta}(z^N)).$$

Для завершения доказательства остается взять нижнюю грань по $x^N\in X^N$ от обеих частей полученного равенства. $\triangle$

9. Наилучшая гарантированная точность в классе последовательных алгоритмов. Рассмотрим класс всех последовательных алгоритмов $\widetilde{A}^N$ (см. (3.11)), считая произвольные итоговые операции $\widetilde{\beta} \in \widetilde{B}_N$ допустимыми. Положим

$$\varepsilon_N = \inf_{x_1 \in X_1} \sup_{y_1 \in \{y_1 | F(z^1) \neq \varnothing\}} \ldots$$
$$\ldots \inf_{x_N \in X_N} \sup_{y_N \in \{y_N | F(z^N) \neq \varnothing\}} \inf_{\beta \in B} \sup_{f \in F(z^N)} \gamma(S(f), \beta) =$$
$$= \inf_{x_1 \in X_1} \sup_{y_1 \in \{y_1 | F(z^1) \neq \varnothing\}} \ldots \inf_{x_N \in X_N} \sup_{y_N \in \{y_N | F(z^N) \neq \varnothing\}} \varepsilon_N(z^N). \tag{4.13}$$

Теорема 4.4. *Справедливо равенство*

$$\inf_{\alpha \in \widetilde{A}^N} \sup_{f \in F} \varepsilon(\alpha, f) = \varepsilon_N.$$

Доказательство. В доказательстве через $\widetilde{x}^{i+1}(y^i)$ будем обозначать вектор, получаемый по формулам (3.2):

$$\widetilde{x}^{i+1}(y^i) = (x_1, x_2, \ldots, x_{i+1}),$$

где $x_1 = \widetilde{x}_1$ — первая компонента $\widetilde{x}^{i+1}$, $x_2 = \widetilde{x}_2(x_1, y_1), \ldots$ $\ldots, x_{i+1} = \widetilde{x}_{i+1}(x_1, \ldots, x_i, y_1, \ldots, y_i)$, при этом y^i предполагается реализуемым при применении алгоритма $\widetilde{x}^{i+1}$.

Выпишем теперь цепочку равенств, которые доказывают теорему (пояснения ниже):

$$\inf_{\alpha \in \widetilde{A}^N} \sup_{f \in F} \varepsilon(\alpha, f) =$$
$$= \inf_{\widetilde{x}^N} \inf_{\widetilde{\beta}} \sup_{y^N} \sup_{f \in F(\widetilde{x}^N(y^{N-1}), y^N)} \gamma(S(f), \widetilde{\beta}(\widetilde{x}^N(y^{N-1}), y^N)) =$$
$$= \inf_{\widetilde{x}^N} \sup_{y^N} \inf_{\beta \in B} \sup_{f \in F(\widetilde{x}^N(y^{N-1}), y^N)} \gamma(S(f), \beta) =$$
$$= \inf_{\widetilde{x}^{N-1}} \inf_{\widetilde{x}_N} \sup_{y^{N-1}} \sup_{y_N} \inf_{\beta \in B} \sup_{f \in F(\widetilde{x}^N(y^{N-1}), y^N)} \gamma(S(f), \beta) =$$
$$= \inf_{\widetilde{x}^{N-1}} \sup_{y^{N-1}} \inf_{x_N \in X_N} \sup_{y_N} \inf_{\beta \in B} \sup_{f \in F(\widetilde{x}^{N-1}(y^{N-2}), x_N, y^N)} \gamma(S(f), \beta) =$$
$$= \ldots = \varepsilon_N.$$

В первом равенстве нижние грани берутся по множеству $\widetilde{X}^N \times \widetilde{B}_N = \widetilde{A}^N$, а верхняя грань по множеству всех векторов y^N, реализуемых при применении алгоритма $\widetilde{x}^N$. Очевидно, $F = \bigcup_{y^N} F(\widetilde{x}^N(y^{N-1}), y^N)$, что оправдывает за-

мену в первом равенстве операции взятия верхней грани по $f \in F$ двумя последовательными операциями взятия верхних граней.

Для обоснования второго равенства в цепочке зафиксируем $\tilde{x}^N$, применим лемму 4.1 к функции

$$g(\tilde{\beta}, y^N) = \sup_{f \in F(\tilde{x}^N(y^{N-1}), y^N)} \gamma(S(f), \tilde{\beta}(\tilde{x}^N(y^{N-1}), y^N))$$

и возьмем нижнюю грань по $\tilde{x}^N$ от обеих частей полученного равенства.

Последующие равенства обосновываются рассуждениями того же типа. △

10. Случай фиксированной итоговой операции. Будем считать, что итоговая операция $\tilde{\beta}$ фиксирована и тем самым выбор алгоритма $\alpha = (\tilde{x}^N, \tilde{\beta})$ сводится к выбору $\tilde{x}^N$. Дело обстоит именно так в случае (3.4), когда допустима единственная итоговая операция $\tilde{\beta}$. Кроме того, в случае, когда допустимы произвольные итоговые операции (см. (3.5)) и при любом реализуемом z^N нижняя грань в (4.3) достигается, т. е. центральная итоговая операция $\tilde{\beta}_*$ существует, можно также зафиксировать $\tilde{\beta} = \tilde{\beta}_*$ (если равенство (4.4) определяет центральную итоговую операцию не единственным образом, то фиксируется любая из них). Как показывают результаты п. 5, мы и в этом случае, фиксируя $\tilde{\beta}$ указанным способом, выбор оптимального алгоритма α сводим к выбору оптимального $\tilde{x}^N$.

Так или иначе, пусть итоговая операция $\tilde{\beta}$ фиксирована. В такой ситуации удобно критерий ε, определенный формулой (1.1) на множестве $\tilde{A}^N \times F$ (см. (3.11)), определить также на множествах $X^N \times F$ (см. (3.8)) и $\tilde{X}^N \times \times F$ (см. (3.10)). Положим

$$\varepsilon(x^N, f) = \gamma(S(f), \tilde{\beta}(x^N, x_1(f), \ldots, x_N(f))),$$

$$\varepsilon(\tilde{x}^N, f) = \varepsilon(x^N, f), \quad (4.14)$$

где во втором равенстве вектор x^N определяется алгоритмом $\tilde{x}^N$ и функцией f по формулам (3.2).

Переформулируем теоремы 4.3, 4.4 таким образом, чтобы наряду со случаем (3.5) они охватывали и случай (3.4) единственной допустимой итоговой операции. Для этого распространим на последний случай введенное

равенством (4.5) обозначение $\varepsilon_N(z^N)$. Именно, положим

$$\varepsilon_N(z^N) = \sup_{f \in (z^N)} \gamma(S(f), \widetilde{\beta}(z^N)), \tag{4.15}$$

где $\widetilde{\beta}$ — зафиксированная нами итоговая операция. Если в случае (3.5) центральная итоговая операция существует, то (4.15) совпадает с (4.3), поскольку мы условились в этом случае в качестве $\widetilde{\beta}$ фиксировать центральную итоговую операцию $\widetilde{\beta}_*$. Нетрудно видеть, что

$$\varepsilon_N(z^N) = \sup_{f \in F(z^N)} \varepsilon(x^N, f). \tag{4.16}$$

С помощью введенных обозначений наилучшая гарантированная точность в классе пассивных алгоритмов X^N (см. (3.8)) может быть записана как $\inf\limits_{x^N \in X^N} \sup\limits_{f \in F} \varepsilon(x^N, f)$, а теорема 4.3 принимает при этом следующий вид.

Теорема 4.5. *Справедливо равенство*

$$\inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f) = \inf_{x^N \in X^N} \sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \varepsilon_N(z^N). \ \triangle$$

Аналогом же теоремы 4.4 выступает следующее утверждение.

Теорема 4.6. *Справедливо равенство*

$$\inf_{\widetilde{x}^N \in \widetilde{X}^N} \sup_{f \in F} \varepsilon(\widetilde{x}^N, f) = \varepsilon_N.$$

Доказательство. В случае (3.5) доказываемое утверждение совпадает с теоремой 4.4, а в случае (3.4) доказательство то же самое, отпадает только необходимость взятия нижних граней по β. $\triangle$

11. Наилучшая гарантированная точность в классах алгоритмов с задержкой информации и блочных алгоритмов. По аналогии с теоремой 4.4 доказываются следующие две теоремы.

Теорема 4.7. *Пусть $\widehat{X}^N$ — множество алгоритмов с задержкой информации, определяемое условиями* (3.13), (3.14). *Тогда*

$$\inf_{\widetilde{x}^N \in \widehat{X}^N} \sup_{f \in F} \varepsilon(\widetilde{x}^N, f) =$$

$$= \inf_{x_1, \ldots, x_{\tau+1}} \sup_{y_1} \inf_{x_{\tau+2}} \sup_{y_2} \ldots \inf_{x_N} \sup_{y_{N-\tau}, \ldots, y_N} \varepsilon_N(z^N),$$

где нижние и верхние грани берутся по таким множествам, что вектор z^N реализуем. △

Теорема 4.8. *Пусть $\widehat{X}^N$ — множество блочных алгоритмов, определяемое условием* (3.15). *Тогда*

$$\inf_{\widetilde{x}^N \in \widehat{X}^N} \sup_{f \in F} \varepsilon(\widetilde{x}^N, f) = \inf_{x_1, \ldots, x_{N_1}} \sup_{y_1, \ldots, y_{N_1}} \ldots$$

$$\ldots \inf_{x_{N_1+\ldots+N_{k-1}+1}, \ldots, x_N} \sup_{y_{N_1+\ldots+N_{k-1}+1}, \ldots, y_N} \varepsilon_N(z^N),$$

где нижние и верхние грани берутся по таким множествам, что вектор z^N реализуем. △

12. Модель вычислений с нефиксированным N. В дальнейшем, наряду с рассматривавшейся до сих пор моделью с фиксированным числом информационных вычислений N, будем рассматривать также ее модификацию на случай нефиксированного N. Пусть

$$\widetilde{x} = (\widetilde{x}_1, \widetilde{x}_2, \ldots, \widetilde{x}_N, \ldots) \tag{4.17}$$

— последовательность отображений вида (3.1). Множество всех таких последовательностей, которые будут теперь выступать в роли алгоритмов, обозначим через $\widetilde{X}$.

Эффективность алгоритмов $\widetilde{x} \in \widetilde{X}$ будем оценивать критерием (4.14), предполагая, что он определен при любом N. Таким образом, если произведено всего N информационных вычислений (N заранее не фиксируется), то в качестве итоговой операции выбирается заранее зафиксированная при каждом N операция $\widetilde{\beta}$.

Будем говорить, что определенный равенством (4.17) алгоритм $\widetilde{x}$ *сходится* на классе F, если

$$\sup_{f \in F} \varepsilon(\widetilde{x}^N, f) \to 0 \quad \text{при} \quad N \to \infty;$$

асимптотически оптимален в $\widetilde{X}$ на классе F, если

$$\sup_{f \in F} \varepsilon(\widetilde{x}^N, f) \sim \varepsilon_N;$$

оптимален по порядку в $\widetilde{X}$ на классе F, если

$$\sup_{f \in F} \varepsilon(\widetilde{x}^N, f) \asymp \varepsilon_N,$$

где $\widetilde{x}^N = (\widetilde{x}_1, \ldots, \widetilde{x}_N)$, а ε_N определено равенствами (4.13), (4.15).

Запись $a_N \sim b_N$ означает, что две сходящиеся к нулю при $N \to \infty$ последовательности $\{a_N\}$, $\{b_N\}$ *эквивалентны*, т. е. $\lim\limits_{N\to\infty} a_N/b_N = 1$ (при этом считаем, что $0/0 = 1$).

Запись $a_N \asymp b_N$ означает, что две сходящиеся к нулю при $N \to \infty$ последовательности $\{a_N\}$, $\{b_N\}$ *имеют одинаковый порядок сходимости*, т. е. существуют N_0, $c_1 > 0$, $c_2 > 0$ такие, что при любом $N \geqslant N_0$

$$c_1|b_N| \leqslant |a_N| \leqslant c_2|b_N|.$$

Пусть имеется последовательность алгоритмов $\{\tilde{x}(N)\}$, $\tilde{x}(N) \in \tilde{X}^N$. Если

$$\sup_{f\in F} \varepsilon(\tilde{x}(N), f) \sim \varepsilon_N \quad \text{или} \quad \sup_{f\in F} \varepsilon(\tilde{x}(N), f) \asymp \varepsilon_N,$$

то последовательность $\{\tilde{x}(N)\}$ называют *асимптотически оптимальной* или, соответственно, *оптимальной по порядку*. Следует, однако, подчеркнуть, что построение асимптотически оптимального или оптимального по порядку алгоритма вида (4.17) из элементов этой последовательности может оказаться невозможным.

13. Оптимальные по числу информационных вычислений алгоритмы. Зафиксируем теперь точность ε, которую необходимо обеспечить при решении задачи. Положим

$$N_\varepsilon = \min\{N \mid \varepsilon_N \leqslant \varepsilon\}. \tag{4.18}$$

Алгоритм α_0 называется *оптимальным по числу информационных вычислений* на классе F среди всех последовательных алгоритмов, обеспечивающих заданную точность ε, если

$$\alpha_0 \in \tilde{A}^{N_\varepsilon} \quad \text{и} \quad \sup_{f\in F} \varepsilon(\alpha_0, f) \leqslant \varepsilon. \tag{4.19}$$

Замечание. Введенные понятия (асимптотически оптимальный, оптимальный по порядку, оптимальный по числу информационных вычислений алгоритмы) можно формулировать не только применительно к множеству всех последовательных алгоритмов, но и применительно к произвольному подмножеству этого множества. △

14. Оптимальные на один шаг алгоритмы. Понятие оптимального на один шаг алгоритма изучалось в рамках минимаксной и иных концепций оптимальности, например, в работах [355] (см. также [26]) для задачи ре-

шения уравнений, в [100, 238, 292, 297] — для задачи поиска экстремума.

Пусть алгоритм $\tilde{x}$ задан равенством (4.17), а его эффективность оценивается по критерию (4.14). Следует задать еще *правило остановки*.

Остановка производится обычно или после заранее заданного числа шагов N, или после гарантированного достижения заданной точности ε, т. е. после j шагов, где (см. (4.16))

$$\varepsilon_{j-1}(z^{j-1}) > \varepsilon, \quad \varepsilon_j(z^j) \leqslant \varepsilon. \tag{4.20}$$

Понятно, что фактическая точность решения задачи для данной функции f может оказаться значительно лучше гарантированной апостериорной точности.

Алгоритм $\tilde{x}$ называется *оптимальным на один шаг* в $\tilde{X}$ на классе F, если

$$\min_{x \in X_1} \sup_{f \in F} \varepsilon(x, f) = \sup_{f \in F} \varepsilon(x_1, f), \tag{4.21}$$

где $x_1 = \tilde{x}_1$, и для любой реализуемой ситуации $z^i = (x^i, y^i)$, $i \geqslant 1$,

$$\min_{x \in X_{i+1}} \sup_{f \in F(z^i)} \varepsilon(x^i, x, f) = \sup_{f \in F(z^i)} \varepsilon(x^i, x^{i+1}, f), \tag{4.22}$$

где $x_{i+1} = \tilde{x}_{i+1}(z^i)$.

Таким образом, оптимальный на один шаг алгоритм гарантирует на каждом шаге максимальное улучшение точности решения задачи.

Несмотря на интуитивную привлекательность понятия и простоту определения оптимальных на один шаг алгоритмов, их использование далеко не во всех случаях может принести успех. Так, например, пусть при поиске экстремума унимодальной функции ([277]) с помощью алгоритмов, использующих лишь значения функции, точка экстремума после i шагов локализована в интервале длины l, причем значение функции уже вычислено в точке, отстоящей на расстояние δ от одного из концов интервала.

Нетрудно видеть, что один из оптимальных на один шаг алгоритмов заключается в том, чтобы на каждом шаге вычислять значение функции в точке, расположенной в интервале локализации экстремума симметрично точке уже проведенного внутри интервала вычисления. После $(i+n)$-го шага этого алгоритма длина интервала

локализации окажется равной $l-n\delta$, т. е. улучшение может быть сколь угодно малым при достаточно малом δ. В то же время применение оптимального пассивного [277, 423], а тем более оптимального последовательного алгоритма [291, 347, 364, 371, 466] даст значительно большее улучшение.

Тем не менее целый ряд обстоятельств привлекает внимание к построению (в рамках различных моделей вычислений) алгоритмов, обладающих свойством оптимальности на один шаг. Важным обстоятельством является возможность перестройки модели вычислений в ходе вычислительного процесса. Например, в ходе вычислений может происходить уточнение некоторых «глобальных» параметров, определяющих функциональный класс (см. п. 2 § 4 гл. 2).

Часто решение о числе информационных вычислений и необходимой абсолютной точности решения задачи приходится принимать в силу тех или иных обстоятельств в ходе вычислений, что также является аргументом в пользу применения оптимальных на один шаг алгоритмов. Такими обстоятельствами могут быть отсутствие априорной информации о сложности вычисления значений оптимизируемой функции, что делает невозможной заблаговременную оценку необходимых вычислительных ресурсов, необходимость отыскания экстремума с заданной относительной погрешностью и др. Наконец, среди побудительных мотивов применения оптимальных на один шаг алгоритмов следует отметить относительную простоту их отыскания и реализации.

§ 5. Сравнение наилучших гарантированных результатов в классах пассивных и последовательных алгоритмов

В этом параграфе будут получены достаточные условия совпадения наилучших гарантированных результатов в классах пассивных и всех последовательных алгоритмов.

1. Постановка задачи. Пусть итоговая операция $\tilde{\beta}$ фиксирована (см. п. 10 § 4). Поскольку множество пассивных алгоритмов X^N является подмножеством множества всех последовательных алгоритмов $\tilde{X}^N$ (см. (3.8), (3.10), (3.12)), то наилучший гарантированный результат

(точность) в $\widetilde{X}^N$ не хуже, чем в X^N, т. е.

$$\inf_{\widetilde{x}^N \in \widetilde{X}^N} \sup_{f \in F} \varepsilon(\widetilde{x}^N, f) \leqslant \inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f), \tag{5.1}$$

где критерий ε определен равенством (4.14).

Для некоторых задач в (5.1) имеет место строгое неравенство, причем левая часть значительно меньше правой. Так, например, в задаче отыскания корня строго монотонной функции, принимающей на концах отрезка [0, 1] значения разных знаков, или в задаче поиска экстремума унимодальной на отрезке [0, 1] функции улучшение зависит от N экспоненциально [276, 364, 371, 372].

В то же время для некоторых других задач в (5.1) имеет место равенство, т. е.

$$\inf_{\widetilde{x}^N \in \widetilde{X}^N} \sup_{f \in F} \varepsilon(\widetilde{x}^N, f) = \inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f). \tag{5.2}$$

Таким образом, возможность использования не только пассивных, но и любых последовательных алгоритмов не позволяет улучшить результат, гарантированный в классе пассивных алгоритмов. Однако, как мы убедимся в дальнейшем, это отнюдь не означает, что применение последовательных алгоритмов нецелесообразно. Применение последовательных алгоритмов при решении конкретных задач может дать значительный эффект и в тех случаях, когда справедливо соотношение (5.2). Этот тезис получит подробное развитие в § 6, где вводится понятие последовательно-оптимального алгоритма, а также в последующих главах, где строятся последовательно-оптимальные алгоритмы для различных задач численного анализа и обсуждаются результаты применения этих алгоритмов. Сам факт справедливости равенства (5.2) представляет не только теоретический интерес, но имеет также и практическое значение, существенно упрощая в ряде случаев построение последовательно-оптимальных алгоритмов.

По-видимому, впервые такого рода результат был получен для задачи интегрирования монотонных функций [372]. Дальнейшее развитие этот вопрос получил в работах [20, 109, 202, 240, 350]. Так, в [20] найдены достаточные условия для выполнения (5.2) в случае, когда S, $x_1 \in X_1, \ldots, x_N \in X_N$ — линейные функционалы. В [240] справедливость (5.2) доказана впервые для некоторого нелинейного функционала.

Здесь мы установим один факт общего характера (теорема 5.1), имеющий полезные приложения, в частности, позволяющий получить результат, аналогичный [20], но при менее ограничительных предположениях (теорема 5.2).

2. Основная теорема. Положим (см. (4.14)—(4.16))

$$\varepsilon_N(x^N, f) \stackrel{\text{def}}{=} \varepsilon_N(x^N, x^N(f)) = \sup_{g \in F(x^N, x^N(f))} \varepsilon(x^N, g), \quad (5.3)$$

где $x^N(f) = (x_1(f), \ldots, x_N(f))$ и, в соответствии с введенным в п. 2 § 3 обозначением, $F(x^N, x^N(f)) = \{g \in F \mid x^N(g) = x^N(f)\}$.

Теорема 5.1 [254]. *Если функция $\varepsilon_N(x^N, f)$ имеет на множестве $X^N \times F$ обобщенную седловую точку, т. е.*

$$\inf_{x^N \in X^N} \sup_{f \in F} \varepsilon_N(x^N, f) = \sup_{f \in F} \inf_{x^N \in X^N} \varepsilon_N(x^N, f), \quad (5.4)$$

то выполняется равенство (5.2), *т. е. наилучшие гарантированные результаты в классах пассивных и всех последовательных алгоритмов совпадают.*

Доказательство. Пусть справедливо равенство (5.4) и v — общее значение его левой и правой частей. В силу (5.3),

$$\sup_{f \in F} \varepsilon_N(x^N, f) = \sup_{f \in F} \varepsilon(x^N, f),$$

поэтому

$$\inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f) = v. \quad (5.5)$$

Далее, из (5.4) следует, что для любого $\delta > 0$ существует функция $f_\delta \in F$ такая, что

$$\inf_{x^N \in X^N} \varepsilon_N(x^N, f_\delta) \geqslant v - \delta,$$

откуда

$$\sup_{f \in F(x^N, x^N(f_\delta))} \varepsilon(x^N, f) = \varepsilon_N(x^N, f_\delta) \geqslant v - \delta, \quad x^N \in X^N. \quad (5.6)$$

Пусть $\tilde{x}^N \in \tilde{X}^N$ — произвольный последовательный алгоритм, а вектор $x_\delta^N \in X^N$ определяется алгоритмом $\tilde{x}^N$ и функцией f_δ по формулам (3.2). Тогда, в силу

определения (4.14) и формулы (5.6),

$$\sup_{f\in F} \varepsilon(\tilde{x}^N, f) \geqslant \sup_{f\in F(x_\delta^N, x_\delta^N(f_\delta))} \varepsilon(x_\delta^N, f) \geqslant v - \delta$$

(действительно, при любой функции $f \in F(x_\delta^N, x_\delta^N(f_\delta))$ алгоритм $\tilde{x}^N$ определяет один и тот же вектор x_δ^N). Поскольку $\tilde{x}^N \in \tilde{X}^N$ и $\delta > 0$ произвольны, то

$$\inf_{\tilde{x}^N\in\tilde{X}^N} \sup_{f\in F} \varepsilon(\tilde{x}^N, f) \geqslant v. \tag{5.7}$$

Справедливость равенства (5.2) следует из (5.1), (5.5), (5.7). △

Из доказанной теоремы легко выводится следующее важное утверждение, впервые полученное в [350] при более жестких ограничениях.

Следствие. *Если существует такая функция* $f_0 \in F$, *что*

$$\varepsilon_N(x^N, f_0) \geqslant \varepsilon_N(x^N, f), \quad x^N \in X^N, \quad f \in F, \tag{5.8}$$

то выполняется равенство (5.2), *т. е. наилучшие гарантированные результаты в классах пассивных и всех последовательных алгоритмов совпадают.*

Доказательство. В силу (5.8)

$$\sup_{f\in F} \inf_{x^N\in X^N} \varepsilon_N(x^N, f) \geqslant \inf_{x^N\in X^N} \varepsilon_N(x^N, f_0) \geqslant$$

$$\geqslant \inf_{x^N\in X^N} \sup_{f\in F} \varepsilon_N(x^N, f).$$

Отсюда следует (5.4), поскольку, как известно,

$$\sup_{f\in F} \inf_{x^N\in X^N} \varepsilon_N(x^N, f) \leqslant \inf_{x^N\in X^N} \sup_{f\in F} \varepsilon_N(x^N, f). \ \triangle$$

3. Задача приближения линейного функционала на выпуклом множестве по линейной информации. Пусть S — функционал, $B = \mathbf{R}$,

$$\gamma(S(f), \beta) = |S(f) - \beta| \tag{5.9}$$

и произвольные итоговые операции считаются допустимыми. Предположим, что в качестве итоговой операции зафиксирована определенная соотношением (4.4) центральная итоговая операция $\tilde{\beta}_*$, т. е. в формулах (4.14)

$$\tilde{\beta} = \tilde{\beta}_*. \tag{5.10}$$

В этом случае в силу определений (5.3) и (4.3) имеем

$$\varepsilon_N(x^N, f) = \inf_{\beta\in\mathbf{R}} \sup_{g\in F(x^N, x^N(f))} |S(g) - \beta| =$$

$$= \frac{1}{2}\left(\sup_{g\in F(x^N, x^N(f))} S(g) - \inf_{g\in F(x^N, x^N(f))} S(g)\right) =$$

$$= \frac{1}{2} \sup_{g_1, g_2\in F(x^N, x^N(f))} (S(g_1) - S(g_2)). \quad (5.11)$$

При решении вопроса о наличии у функции $\varepsilon_N(x^N, f)$ седловой точки может оказаться полезным следующий факт, установленный в [350] при более ограничительных предположениях. В его формулировке подразумевается, что $Y_1, \ldots, Y_N$ — линейные пространства.

Лемма 5.1. *Если выполняются предположения* (5.9), (5.10), *множество* F *выпукло*, $x_i \in X_i$ — *линейные операторы*, $i = 1, \ldots, N$, S — *линейный функционал, то функция* $\varepsilon_N(x^N, f)$ *выпукла по* f *на* F.

Доказательство. Пусть $f_1, f_2 \in F$. Тогда для любого $\delta > 0$, в силу (5.11), существуют такие f_i', f_i'', что

$$x^N(f_i') = x^N(f_i'') = x^N(f_i),$$

$$\varepsilon_N(x^N, f_i) \leqslant \frac{1}{2}(S(f_i'') - S(f_i')) + \delta, \quad i = 1, 2.$$

Замечая теперь, что вследствие линейности x^N

$$x^N(\xi f_1'' + (1-\xi) f_2'') = x^N(\xi f_1' + (1-\xi) f_2') =$$
$$= x^N(\xi f_1 + (1-\xi) f_2),$$

получаем при произвольном $\xi \in [0, 1]$

$$\varepsilon_N(x^N, \xi f_1 + (1-\xi) f_2) \geqslant$$
$$\geqslant \frac{1}{2} S(\xi f_1'' + (1-\xi) f_2'') - \frac{1}{2} S(\xi f_1' + (1-\xi) f_2') =$$
$$= \frac{1}{2}\xi(S(f_1'') - S(f_1')) + \frac{1}{2}(1-\xi)(S(f_2'') - S(f_2')) \geqslant$$
$$\geqslant \xi\varepsilon_N(x^N, f_1) + (1-\xi)\varepsilon_N(x^N, f_2) - \delta,$$

откуда, в силу произвольности $\delta > 0$, вытекает утверждение леммы. △

Получим теперь с помощью следствия из теоремы 5.1 основную теорему данного пункта.

Теорема 5.2. *Пусть выполняются предположения* (5.9), (5.10), F — *выпуклое центрально-симметричное множество с центром* f_0, *множества* X_i *состоят из линейных операторов*, $i=1, \ldots, N$, *и функционал* S *также линеен. Тогда выполняется равенство* (5.2), *т. е. наилучшие гарантированные результаты в классах пассивных и всех последовательных алгоритмов совпадают, и любой оптимальный в* X^N *алгоритм оптимален, таким образом, и в* $\tilde{X}^N$.

Доказательство. Зафиксируем произвольные $f \in F$, $x^N \in X^N$. Пусть

$$g_1,\ g_2 \in F(x^N,\ x^N(f)). \tag{5.12}$$

Тогда $2f_0 - g_1 \in F$, $2f_0 - g_2 \in F$ в силу симметрии множества F и

$$f_1 \stackrel{\text{def}}{=} \frac{1}{2}(g_1 + 2f_0 - g_2) \in F, \quad f_2 \stackrel{\text{def}}{=} \frac{1}{2}(g_2 + 2f_0 - g_1) \in F$$

в силу выпуклости F. Кроме того, $x^N(f_1) = x^N(f_2) = x^N(f_0)$, откуда, учитывая (5.11), получаем

$$\varepsilon_N(x^N, f_0) \geqslant \frac{1}{2}\ (S(f_1) - S(f_2)) = \frac{1}{2}(S(g_1) - S(g_2)).$$

Так как g_1, g_2 — любые функции, удовлетворяющие (5.12), то в силу (5.11)

$$\varepsilon_N(x^N,\ f_0) \geqslant \varepsilon_N(x^N,\ f).$$

Доказательство завершается теперь применением следствия из теоремы 5.1. △

4. Задача глобальной оптимизации. Пусть рассматривается задача глобальной оптимизации (1.4), т. е.

$$S(f) = \sup_{x \in K} f(x).$$

Вопрос о совпадении наилучших гарантированных результатов в классах пассивных и всех последовательных алгоритмов был, по-видимому впервые для такой задачи, рассмотрен в [240]. В дальнейшем теми же методами данный вопрос решен в [109] для более общих функциональных классов (см. также [202]).

Здесь мы ограничимся одним простым результатом, который устанавливается с помощью рассуждений, отличных от тех, что используются в [109, 240].

Пусть $x_i(f) = f(x_i)$, $x_i \in K$, $i = 1, \ldots, N$, и допустима единственная итоговая операция (см. (3.6))

$$\widetilde{\beta}(z^N) = \max_{i=1,\ldots,N} y_i.$$

В этом случае в силу (4.14)

$$\varepsilon(x^N, f) = \sup_{x \in K} f(x) - \max_{i=1,\ldots,N} f(x_i). \tag{5.13}$$

Теорема 5.3. *Пусть функциональный класс F содержит константы $c \in \mathbf{R}$ и обладает свойствами*

$$f \in F \Rightarrow f + c \in F, \quad c \in \mathbf{R}, \tag{5.14}$$

$$f \in F \Rightarrow f_+ \stackrel{\text{def}}{=} \max\{f, 0\} \in F. \tag{5.15}$$

Тогда для задачи глобальной оптимизации с критерием (5.13) *выполняется равенство* (5.2), *т. е. наилучшие гарантированные результаты в классах пассивных и всех последовательных алгоритмов совпадают.*

Доказательство. Пусть $x^N \in X^N = K^N$, $g \in F$, тогда в силу (5.14), (5.15)

$$\bar{g} \stackrel{\text{def}}{=} \Big(g - \max_{i=1,\ldots,N} g(x_i)\Big)_+ \in F, \; x^N(\bar{g}) = x^N(0) = 0$$

и (см. (5.3))

$$\varepsilon(x^N, g) = \varepsilon(x^N, \bar{g}) \leqslant \varepsilon_N(x^N, 0).$$

Зафиксировав произвольную функцию $f \in F$ и взяв в полученном неравенстве верхнюю грань по $g \in F(x^N, x^N(f))$, получим, что $\varepsilon_N(x^N, f) \leqslant \varepsilon_N(x^N, 0)$. Доказательство завершается теперь применением следствия из теоремы 5.1. △

Следствие. *Для задачи поиска глобального экстремума функции из заданного произвольной квазиметрикой ρ функционального класса F_ρ* (*см.* § 2) *наилучшие гарантированные результаты в классах пассивных и всех последовательных алгоритмов совпадают, т. е. справедливо равенство* (5.2), *и любой оптимальный в X^N алгоритм оптимален, таким образом, и в $\widetilde{X}^N$.*

Доказательство. Очевидно, что класс F_ρ содержит константы, обладает свойством (5.14), а в силу леммы 2.7 — и свойством (5.15). △

В заключение отметим, что доказательство теорем 5.2, 5.3 получено с помощью следствия из теоремы 5.1. Однако в ряде случаев совпадение минимакса погрешности в классах пассивных и последовательных алгоритмов

удается доказать с помощью теоремы 5.1, в то время как универсальной наихудшей функции f_0 не существует (т. е. следствие из теоремы 5.1 не работает). Примером может служить задача интегрирования в классах монотонных функций и функций, удовлетворяющих условию Липшица (в обоих случаях предполагается, что функции заданы на отрезке числовой прямой, а их значения на концах отрезка фиксированы и различны, см. [259]).

§ 6. Последовательно-оптимальные алгоритмы

В этом параграфе определяется и обсуждается центральное в книге понятие последовательно-оптимального алгоритма.

1. Примеры, показывающие необходимость уточнения понятия оптимальности. Начнем с того, что для различных задач численного анализа и различных функциональных классов приведем примеры таких ситуаций и таких оптимальных алгоритмов, которые предписывают производить явно не наилучшие в данных ситуациях действия. При этом мы вынуждены несколько забегать вперед, используя устанавливаемый в последующих главах факт оптимальности упомянутых алгоритмов (примеры 1—3).

Пример 1. Пусть F — класс функций, удовлетворяющих на отрезке [0, 1] условию Липшица с заданной константой M; каждое информационное вычисление (в этом и в следующих трех примерах) — это вычисление значения функции в некоторой точке. В соответствии с теоремой 2.1 и формулой (3.2) гл. 2 составная квадратурная формула прямоугольников с узлами $1/(2N)$, $3/(2N)$, ..., $(2N-1)/(2N)$ оптимальна. В силу теоремы 1.3 гл. 2 эта квадратура является оптимальным по точности пассивным алгоритмом численного интегрирования. Применяя этот алгоритм, за приближение к интегралу $\int_0^1 f(x)\,dx$ следует принять величину $\frac{1}{N}\sum_{i=1}^{N} f\left(\frac{2i-1}{2N}\right)$. Согласно теореме 5.2 и лемме 2.8 данный алгоритм оптимален также в классе всех последовательных алгоритмов. Поскольку порядок узлов не имеет значения, предположим, что значения функции f вычисляются последовательно в узлах

$$\frac{1}{2N},\quad \frac{2N-1}{2N},\quad \frac{3}{2N},\quad \frac{5}{2N},\quad \ldots,\quad \frac{2N-3}{2N} \tag{6.1}$$

и на первых двух шагах получены результаты

$$f\left(\frac{1}{2N}\right) = M\frac{1}{2N}, \quad f\left(\frac{2N-1}{2N}\right) = M\frac{2N-1}{2N}. \tag{6.2}$$

Нетрудно видеть, что в этом случае $f(x) = Mx$ при $x \in [1/(2N), (2N-1)/(2N)]$. Таким образом, оптимальный алгоритм предписывает проводить оставшиеся $N-2$ вычисления в точках, значения функции f в которых уже известны вычислителю, что явно бессмысленно. △

Пример 2. Для того же, что и в примере 1, класса функций рассмотрим задачу глобальной оптимизации. Оптимальный пассивный алгоритм ее решения состоит в переборе значений функции по сетке (6.1); этот же алгоритм оптимален в классе всех последовательных алгоритмов (см. замечания 2, 3 к теореме 2.1 гл. 4). Если получены результаты (6.2), то, как и в предыдущем примере, проводить оставшиеся $N-2$ вычисления, используя данный алгоритм, бессмысленно. △

Пример 3. Пусть F — класс не убывающих на отрезке [0, 1] функций с фиксированными на концах отрезка значениями: $f(0) = 0$, $f(1) = 1$. Как показано в п. 6 § 3 гл. 2 (теорема 3.4), составная квадратурная формула трапеций с узлами $0, 1/(N+1), 2/(N+1), \ldots, N/(N+1), 1$ является оптимальным по точности пассивным алгоритмом численного интегрирования. Там же показано, что этот алгоритм оптимален в классе всех последовательных алгоритмов. Пусть

$$f\left(\frac{1}{N+1}\right) = 1. \tag{6.3}$$

Очевидно, что в этом случае $f(x) = 1$ при $x \in [1/(N+1), 1]$ и проводить оставшиеся вычисления в точках $2/(N+1), \ldots, N/(N+1)$ не имеет смысла. △

Прежде чем привести последний пример и провести обсуждение, из которого будет следовать необходимость уточнить понятие оптимальности, отметим, что примеры 1—3 позволяют сделать еще одно важное заключение. Из их рассмотрения ясно, что в благоприятных ситуациях, в частности в ситуациях (6.2), (6.3), с помощью последовательных алгоритмов можно обеспечить достижение более высокой точности, чем с помощью пассивных алгоритмов, даже если наилучшие гарантированные результаты в классах пассивных и последовательных алгоритмов совпадают, т. е. справедливо равенство (5.2).

Пример 4. Пусть рассматривается задача поиска максимума унимодальной на отрезке [0, 1] функции. Как известно, N-шаговый ε-оптимальный по точности последовательности алгоритм [276, 277, 364, 371] при $N=4$ заключается в вычислении значений функции f в точках $x_1=2/5$, $x_2=3/5$,

$$x_3=\begin{cases}1/5, & \text{если } f(2/5)>f(3/5),\\ 4/5, & \text{если } f(2/5)<f(3/5),\\ 1/5 \text{ или } 4/5, & \text{если } f(2/5)=f(3/5);\end{cases}$$

наконец, если, к примеру, $f(3/5)<f(2/5)<f(1/5)$, то $x_4=1/5+\varepsilon$. Этот алгоритм гарантирует точность $1/5+\varepsilon$. Однако если после двух шагов оказалось, что

$$f(2/5)=f(3/5), \tag{6.4}$$

то можно гарантировать точность $1/10+\varepsilon$,— достаточно положить $x_3=1/2$, $x_4=1/2+\varepsilon$. △

Приведенные примеры показывают, что данное в § 4 определение оптимального алгоритма не отражает некоторых важных аспектов процесса организации вычислений. Гарантируя наилучшую возможную точность для «наихудшей» функции из класса F, оптимальный по точности алгоритм, вообще говоря, никак не использует выявленные в ходе решения задачи свойства функции, оказавшиеся благоприятными (не наихудшими) для вычислителя. Так, явно не наихудшими (в действительности — наилучшими) являются в примерах 1—4 функции со свойствами (6.2)—(6.4).

Такого рода примеры служат иногда основанием для критики минимаксного подхода (принципа наилучшего гарантированного результата [62]), утверждений о его «перестраховочности», о том, что наименее благоприятные условия, на учет которых ориентирован минимаксный подход, редко встречаются на практике. Однако на самом деле причиной того, что определенные в рамках минимаксного подхода оптимальные алгоритмы не всегда должным образом используют информацию о задаче, получаемую в ходе ее решения, является как раз неполное, не до конца последовательное соблюдение принципа наилучшего гарантированного результата при определении оптимальных алгоритмов. Примеры 1—4 приводят к выводу о необходимости уточнить сформулированное в § 4 понятие оптимальности. Ниже мы даем такое уточнение. Оно заключается в выделении из множества всех опти-

мальных алгоритмов важного подмножества последовательно-оптимальных алгоритмов.

2. Понятие последовательно-оптимального алгоритма. Последовательно руководствуясь стремлением к наилучшим гарантированным результатам, следует стремиться найти не произвольный оптимальный алгоритм, а тот из них, который на каждом шаге[1]) наилучшим образом (в целях достижения наивысшей возможной точности при заданном числе шагов или в целях максимального сокращения числа оставшихся шагов, необходимых для достижения заданной точности) использует накопленную на предыдущих шагах информацию о решаемой задаче (независимо от того, является ли информация наихудшей для вычислителя или нет). Такие алгоритмы мы называем последовательно-оптимальными (формальные определения даются ниже).

Понятие последовательно-оптимального алгоритма было введено в [241] (в этой работе употребляется термин «наилучшая стратегия»). Следует отметить, что для задачи поиска экстремума последовательно-оптимальные алгоритмы для некоторых классов функций были фактически построены, например, в работах [291, 466], хотя в этих работах ставилась лишь цель построения оптимального в рамках минимаксной концепции алгоритма. В идейном плане понятие последовательно-оптимального алгоритма тесно связано со схемами последовательного анализа [25, 26, 36, 37, 169], опирающимися на идеи А. А. Маркова.

По-видимому, последовательно-оптимальные алгоритмы для задачи оптимального восстановления функций были впервые построены в работах [244—246], а для задачи численного интегрирования — в [441, 442].

Построение последовательно-оптимального алгоритма — задача существенно более сложная, чем построение просто оптимального алгоритма. Для ее решения, как мы увидим ниже, приходится, по существу, на каждом шаге решать задачу построения оптимального алгоритма на подклассе исходного класса, определенном фиксацией полученной к данному шагу информации.

Следует отметить, что не для всех задач практическим нуждам оптимизации вычислительного процесса

[1]) Напомним, что шагом алгоритма мы называем проведение очередной пары алгоритмического и информационного вычислений (см. п. 1 § 3).

отвечает понятие только последовательно-оптимального, а не просто оптимального алгоритма. Так, для задачи поиска экстремума унимодальной функции единственным случаем благоприятного для вычислителя поведения функции является точное равенство между значением функции, полученным при очередном вычислении, и одним из ранее вычисленных значений. Случай такого равенства — в некотором смысле исключительная ситуация.

3. Формальные определения. Будем считать, что итоговая операция $\tilde{\beta}$ фиксирована ($\tilde{\beta}$ — либо единственная допустимая, либо центральная итоговая операция), т. е. будем пользоваться критерием (4.14). Положим для $i \geqslant 1$

$$\varepsilon(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f) = \varepsilon(x^i, x_{i+1}, \ldots, x_N, f), \quad (6.5)$$

где $x_{i+1}, \ldots, x_N$ определены при фиксированном x^i отображениями $\tilde{x}_{i+1}, \ldots, \tilde{x}_N$ и функцией f по формулам (3.2). Совокупность отображений $(\tilde{x}_{i+1}, \ldots, \tilde{x}_N)$ мы будем так же, как и совокупность $(\tilde{x}_1, \ldots, \tilde{x}_N)$, называть алгоритмом. Напомним, что $\tilde{X}^N$ обозначает множество всех последовательных алгоритмов.

Определение 1. Алгоритм $\tilde{x}_0^N = (\tilde{x}_1^0, \ldots, \tilde{x}_N^0) \in \tilde{X}^N$ называется *последовательно-оптимальным по точности* на классе F, если он оптимален по точности в $\tilde{X}^N$ на классе F и для любой реализуемой ситуации z^i (см. п. 2 § 3), $1 \leqslant i \leqslant N-1$,

$$\sup_{f \in F(z^i)} \varepsilon(x^i, \tilde{x}_{i+1}^0, \ldots, \tilde{x}_N^0, f) = \min_{\tilde{x}_{i+1}, \ldots, \tilde{x}_N} \sup_{f \in F(z^i)} \varepsilon(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f), \quad (6.6)$$

где минимум берется по всевозможным наборам отображений $(\tilde{x}_{i+1}, \ldots, \tilde{x}_N)$ вида (3.1). △

По аналогии и в полном соответствии с (4.13), (4.15) и (4.16) введем обозначение

$$\varepsilon_N(z^i) = \inf_{x_{i+1} \in X_{i+1}} \sup_{y_{i+1} \in \{y_{i+1} \mid F(z^{i+1}) \neq \varnothing\}} \ldots$$
$$\ldots \inf_{x_N \in X_N} \sup_{y_N \in \{y_N \mid F(z^N) \neq \varnothing\}} \sup_{f \in F(z^N)} \gamma(S(f), \tilde{\beta}(z^N)) =$$
$$= \inf_{x_{i+1} \in X_{i+1}} \sup_{y_{i+1} \in \{y_{i+1} \mid F(z^{i+1}) \neq \varnothing\}} \ldots$$
$$\ldots \inf_{x_N \in X_N} \sup_{y_N \in \{y_N \mid F(z^N) \neq \varnothing\}} \varepsilon_N(z^N). \quad (6.7)$$

Применяя теорему 4.6 к функциональному классу $F(z^i)$ и критерию $\varepsilon(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f)$, получим, что

$$\inf_{\tilde{x}_{i+1},\ldots,\tilde{x}_N} \sup_{f\in F(z^i)} \varepsilon(x^i, \tilde{x}_{i+1},\ldots, \tilde{x}_N, f) = \varepsilon_N(z^i). \qquad (6.8)$$

Сравнивая (6.6), (6.7) и (6.8), убеждаемся, что определение 1 эквивалентно следующему определению.

Определение 2. Алгоритм $\tilde{x}_0^N = (\tilde{x}_1^0, \ldots, \tilde{x}_N^0) \in \tilde{X}^N$ называется последовательно-оптимальным по точности на классе F, если внешняя нижняя грань в выражении (4.13) для ε_N достигается при $x_1 = \tilde{x}_1^0$, а внешняя нижняя грань в выражении (6.7) для $\varepsilon_N(z^i)$ при любой реализуемой ситуации z^i, $1 \leqslant i \leqslant N-1$, достигается при $x_{i+1} = \tilde{x}_{i+1}^0(z^i)$. △

Аналогично понятию ε-оптимального алгоритма можно определить понятие ε-последовательно-оптимального алгоритма.

Ясно, что последовательно-оптимальный алгоритм может оказаться и, как мы убедимся в дальнейшем, в реальных задачах часто оказывается не единственным. В этом случае тот или иной последовательно-оптимальный алгоритм целесообразно выбирать в каждой конкретной ситуации с учетом специфики решаемой задачи. Сделанное замечание относится и к определяемым в следующем пункте алгоритмам, последовательно-оптимальным по числу информационных вычислений.

4. Последовательно-оптимальные по числу информационных вычислений алгоритмы. Зафиксируем теперь точность ε, которую необходимо обеспечить при решении задачи. По аналогии с определенной формулой (4.18) величиной N_ε определим для произвольной реализуемой ситуации z^i величину $N_\varepsilon(z^i)$ формулой

$$N_\varepsilon(z^i) = \min\{N \mid \varepsilon_{i+N}(z^i) \leqslant \varepsilon\}, \qquad (6.9)$$

где $\varepsilon_{i+N}(z^i)$ определено посредством (6.7).

Будем считать, что вычислитель использует правило остановки (4.20), т. е. остановка производится после j шашов в ситуации z^j, если (см. (4.15), (4.16))

$$\varepsilon_{j-1}(z^{j-1}) > \varepsilon, \quad \varepsilon_j(z^j) \leqslant \varepsilon.$$

Пусть по-прежнему итоговая операция $\tilde{\beta}$ фиксирована ($\tilde{\beta}$ — либо единственная допустимая, либо центральная итоговая операция).

Алгоритм $\tilde{x}_0 = (\tilde{x}_1^0, \tilde{x}_2^0, \ldots)$ называется *последовательно-оптимальным по числу информационных вычислений* на классе F при фиксированной точности ε, если он оптимален на классе F по числу информационных вычислений, т. е.

$$\tilde{x}_0 \in \tilde{X}^{N_\varepsilon}, \quad \sup_{f \in F} \varepsilon(\tilde{x}_0, f) \leqslant \varepsilon, \tag{6.10}$$

и для любой реализуемой ситуации z^i, $i \geqslant 1$,

$$\sup_{f \in F(z^i)} \varepsilon\left(x^i, \tilde{x}_{i+1}^0, \ldots, \tilde{x}_{i+N_\varepsilon(z^i)}^0, f\right) \leqslant \varepsilon. \tag{6.11}$$

Очевидно, что если последовательность ситуаций $\{z^i\}$ получена при решении некоторой задачи с помощью последовательно-оптимального алгоритма, то

$$N_\varepsilon(z^1) \leqslant N_\varepsilon - 1, \quad N_\varepsilon(z^{i+1}) \leqslant N_\varepsilon(z^i) - 1, \quad i \geqslant 1. \tag{6.12}$$

В дальнейшем мы убедимся, что последовательно-оптимальные алгоритмы оказываются часто значительно эффективнее оптимальных (это предположение можно сделать и на основе приведенных в начале параграфа примеров 1—3). В таких случаях недостаточно ограничиться нахождением априорных оценок эффективности алгоритмов на классах функций (подобные оценки могут создать ложное представление о безнадежности задачи) и построением на этой основе одного из оптимальных алгоритмов. Тем более нельзя ограничиться построением оптимальных по порядку или асимптотических оптимальных алгоритмов.

§ 7. Стохастические алгоритмы

Стохастические (вероятностные, рандомизированные, статистические) алгоритмы находят довольно широкое применение при решении различных задач численного анализа. Упомянем для примера задачи численного интегрирования [13, 14, 16, 22, 95, 232]; решения уравнений [44]; оптимизации [103, 123, 186, 216—219, 260, 281, 283, 284, 294, 302, 305, 326, 351, 385, 443], в том числе задачи планирования эксперимента [183, 285, 287], где также рассматриваются рандомизированные процедуры. Примыкают к стохастическим и алгоритмы типа [231, 233, 234, 404, 405, 435].

Значительно реже ставятся и решаются вопросы об оценке эффективности стохастических алгоритмов и построении оптимальных стохастических алгоритмов (см., например, работы [13, 14, 16, 22, 44, 186, 240, 248, 250, 278—284, 301—303, 350]).

Следует отметить, что применение стохастических алгоритмов, сравнение их между собой по эффективности, стремление к отысканию оптимальных стохастических алгоритмов предполагает готовность вычислителя пользоваться осредненными критериями оценки эффективности. Такая готовность проявляется чаще при необходимости многократного использования процедуры рандомизации в процессе вычислений.

В этом параграфе описаны различные классы стохастических алгоритмов, дается ряд определений конкретных понятий оптимальности стохастических алгоритмов в рамках минимаксной концепции оптимальности. Однако для того чтобы прояснить вопрос о целесообразности использования стохастических алгоритмов, мы начнем с обсуждения некоторых аспектов применения детерминированных алгоритмов.

1. Процесс решения задачи как антагонистическая игра. Опишем процесс решения задачи как многошаговую антагонистическую игру вычислителя (минимизирующий игрок) и природы (максимизирующий игрок). Пусть для определенности рассматривается случай, когда вычислитель может использовать любой алгоритм из $\bar{A}^N$ (см. (3.11)). В этом случае игра состоит в следующем.

На первом шаге минимизирующий игрок выбирает $x_1 \in X_1$; а максимизирующий, зная x_1, выбирает $y_1 \in \{y_1 | F(z^1) \neq \varnothing\}$. На втором шаге минимизирующий игрок, зная x_1, y_1, выбирает $x_2 \in X_2$, а максимизирующий игрок, зная x_1, x_2, y_1, выбирает $y_2 \in \{y_2 | F(z^2) \neq \varnothing\}$ и так N раз. Наконец, на $(N+1)$-м шаге минимизирующий игрок, зная z^N, выбирает $\beta \in B$, а максимизирующий игрок, зная z^N, β, выбирает $f \in F(z^N)$. Выигрыш максимизирующего (проигрыш минимизирующего) игрока равен $\gamma(S(f), \beta)$. Схематически игру можно представить очередностью выбора альтернатив игроками:

$$x_1,\ y_1,\ x_2,\ y_2,\ \ldots,\ x_N,\ y_N,\ \beta,\ f. \tag{7.1}$$

Нетрудно видеть, что это — игра с полной информацией. Однако у каждого из игроков в каждой позиции имеется, вообще говоря, бесконечное (даже несчетное)

множество альтернатив. Поэтому непосредственно применить известную в теории игр теорему Цермело о наличии в игре с полной информацией седловой точки [149, 289] нельзя.

Все же, в предположении о достижимости всех нижних и верхних граней в выражении (4.13) для ε_N, которое при сделанном предположении можно записать в виде

$$\varepsilon_N = \min_{x_1 \in X_1} \max_{y_1 \in \{y_1 \mid F(z^1) \neq \varnothing\}} \ldots$$

$$\ldots \min_{x_N \in X_N} \max_{y_N \in \{y_N \mid F(z^N) \neq \varnothing\}} \min_{\beta \in B} \max_{f \in F(z^N)} \gamma(S(f), \beta), \quad (7.2)$$

игра (7.1) имеет седловую точку.

Действительно, минимизирующий игрок обеспечивает, согласно теореме 4.4, проигрыш не более ε_N применением любой оптимальной стратегии (существование оптимальной, более того, последовательно-оптимальной стратегии, следует из предположения о достижимости минимумов в (7.2)). С другой стороны, максимизирующий игрок обеспечивает выигрыш не менее ε_N с помощью стратегии, в которой на каждом шаге выбор альтернативы реализует соответствующий максимум в (7.2).

В случае, когда в распоряжении вычислителя находятся лишь пассивные алгоритмы, т. е. множеством стратегий минимизирующего игрока является множество A^N (см. (3.9)), игра представима следующей схемой очередности выбора альтернатив игроками:

$$x^N, \ y^N, \ \beta, \ f. \quad (7.3)$$

В предположении о достижимости минимумов и максимумов в выражении (см. теорему 4.3)

$$\min_{x^N \in X^N} \max_{y^N \in \{y^N \mid F(z^N) \neq \varnothing\}} \min_{\beta \in B} \max_{f \in F(z^N)} \gamma(S(f), \beta) \quad (7.4)$$

игра (7.3) имеет седловую точку (доказательство проводится так же, как и в случае (7.1)).

Если допустима единственная итоговая операция $\tilde{\beta}$ то в выражениях (7.2), (7.4) следует опустить минимум по $\beta \in B$ и заменить $\gamma(S(f), \beta)$ на $\gamma(S(f), \tilde{\beta}(z^N))$, а также опустить β в (7.1), (7.3).

Аналогично можно было бы рассмотреть игры, соответствующие случаям, когда вычислитель применяет алгоритмы с задержкой информации и блочные алгоритмы

Использование наличия седловой точки в игре, описывающей процесс решения задачи, помогает в ряде случаев построить оптимальные и последовательно-оптимальные детерминированные алгоритмы вычислений. Типичный пример такого рода рассуждений имеется в § 2, 3 гл. 3.

В то же время, при рассмотрении стохастических алгоритмов вычислений трактовка процесса решения задачи как игры с полной информацией может не отражать адекватно реального положения вещей и привести к неверному выводу о нецелесообразности применения стохастических алгоритмов.

Рассуждения, приводящие к такому выводу, сводятся к следующему: игра с полной информацией имеет седловую точку в чистых стратегиях и, следовательно, наилучший гарантированный результат при применении смешанных стратегий (стохастических алгоритмов) совпадает с наилучшим гарантированным результатом при применении чистых стратегий (детерминированных алгоритмов), а оптимальный детерминированный алгоритм оптимален в классе всех стохастических алгоритмов.

Ошибочность этих рассуждений состоит в том, что наилучший гарантированный при применении стохастических алгоритмов результат вычислителя в исходной задаче следует определять иначе, нежели наилучший гарантированный в смешанных стратегиях результат в соответствующей антагонистической игре.

Поясним сказанное простейшим примером. Пусть в распоряжении вычислителя находятся лишь пассивные алгоритмы $x^N \in X^N$ и предполагается, что фиксирована некоторая итоговая операция $\tilde{\beta}$. В этом случае соответствующая игра с полной информацией еще проще, чем игра (7.3), и представима следующей схемой очередности выбора альтернатив игроками:

$$x^N,\ f. \tag{7.5}$$

При этом функцией выигрыша служит определенный формулой (4.14) критерий $\varepsilon(x^N, f)$.

Стохастические пассивные алгоритмы (смешанные пассивные стратегии) задаются вероятностными мерами $\sigma^N \in \Sigma^N$, где Σ^N — множество всех вероятностных мер на некоторой σ-алгебре пространства элементарных событий X^N. Применение стохастического пассивного алгоритма σ^N состоит в выборе случайным образом в соответствии с вероятностным распределением σ^N некоторого

$x^N \in X^N$ и затем, так же как и в случае детерминированных алгоритмов, вычислении $y_1 = x_1(f), \ldots, y_N = x_N(f)$, $\beta = \tilde{\beta}(x_1, \ldots, x_N, y_1, \ldots, y_N)$.

Наилучший гарантированный в смешанных стратегиях результат в игре (7.5) есть

$$\inf_{\sigma^N \in \Sigma^N} \int_{X^N} \sup_{f \in F} \varepsilon(x^N, f)\, \sigma^N\{dx^N\} = \inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f). \quad (7.6)$$

Однако в исходной задаче нет никаких оснований считать, что «природа узнáет» конкретную реализацию случайного выбора вычислителя. Поэтому *наилучшим гарантированным при применении стохастических алгоритмов результатом* вычислителя в исходной задаче следует считать величину

$$\inf_{\sigma^N \in \Sigma^N} \sup_{f \in F} \int_{X^N} \varepsilon(x^N, f)\, \sigma^N\{dx^N\}. \quad (7.7)$$

Отметим, что функции, стоящие под знаками интеграла в (7.6), (7.7) и в любой из последующих формул данного параграфа, предполагаются интегрируемыми. Теперь мы можем перейти к определению различных понятий оптимальности для стохастических алгоритмов.

2. Оптимальные по точности стохастические пассивные алгоритмы. Алгоритм $\sigma_0^N \in \Sigma^N$ называется *оптимальным по точности в множестве стохастических пассивных алгоритмов* Σ^N на классе F, если

$$\sup_{f \in F} \int_{X^N} \varepsilon(x^N, f)\, \sigma_0^N\{dx^N\} = \min_{\sigma^N \in \Sigma^N} \sup_{f \in F} \int_{X^N} \varepsilon(x^N, f)\, \sigma^N\{dx^N\}. \quad (7.8)$$

В случае, когда допустимы произвольные итоговые операции и вычислитель может рандомизировать выбор β, стохастический пассивный алгоритм задается парой $(\sigma^N, \tilde{\tau})$, где $\sigma^N \in \Sigma^N$,

$$\tilde{\tau}\colon \{z^N \mid F(z^N) \neq \varnothing\} \to T, \quad (7.9)$$

T — множество всех вероятностных мер на некоторой σ-алгебре пространства элементарных событий B. Множество всех отображений вида (7.9) обозначим через $\tilde{T}$.

Применение стохастического пассивного алгоритма $(\sigma^N, \tilde{\tau}) \in \Sigma^N \times \tilde{T}$ к некоторой функции $f \in F$ состоит в

выборе случайным образом в соответствии с вероятностным распределением σ^N некоторого $x^N \in X^N$, вычислении $y^N = x^N(f) = (x_1(f), \ldots, x_N(f))$, определении $\tau = \tilde{\tau}(z^N) \in T$ и, наконец, выборе случайным образом в соответствии с вероятностным распределением τ некоторого $\beta \in B$.

Результат, гарантированный алгоритмом $(\sigma^N, \tilde{\tau})$ на классе F, равен величине

$$\sup_{f \in F} \int_{X^N} \sigma^N \{dx^N\} \int_B \tilde{\tau}(x^N, x^N(f)) \{d\beta\} \gamma(S(f), \beta).$$

Теперь естественно ввести следующее определение.

Алгоритм $(\sigma_0^N, \tilde{\tau}_0) \in \Sigma^N \times \tilde{T}$ называется *оптимальным по точности в множестве стохастических пассивных алгоритмов* $\Sigma^N \times \tilde{T}$ на классе F, если

$$\sup_{f \in F} \int_{X^N} \sigma_0^N \{dx^N\} \int_B \tilde{\tau}_0(x^N, x^N(f)) \{d\beta\} \gamma(S(f), \beta) =$$

$$= \min_{(\sigma^N, \tilde{\tau}) \in \Sigma^N \times \tilde{T}} \sup_{f \in F} \int_{X^N} \sigma^N \{dx^N\} \int_B \tilde{\tau}(x^N, x^N(f)) \{d\beta\} \gamma(S(f), \beta).$$

По аналогии с теоремой 4.3 можно показать, что для наилучшего гарантированного при применении алгоритмов из $\Sigma^N \times \tilde{T}$ на классе F результата справедливо равенство

$$\inf_{(\sigma^N, \tilde{\tau}) \in \Sigma^N \times \tilde{T}} \sup_{f \in F} \int_{X^N} \sigma^N \{dx^N\} \int_B \tilde{\tau}(x^N, x^N(f)) \{d\beta\} \gamma(S(f), \beta) =$$

$$= \inf_{\sigma^N \in \Sigma^N} \sup_{f_1 \in F} \int_{X^N} \sigma^N \{dx^N\}$$

$$\inf_{\tau \in T} \sup_{f_2 \in F(x^N, x^N(f_1))} \int_B \tau \{d\beta\} \gamma(S(f_2), \beta). \quad (7.10)$$

3. Оптимальные по точности стохастические последовательные алгоритмы. *Стохастический последовательный алгоритм* в случае фиксированной итоговой операции $\tilde{\beta}$ определяется как совокупность отображений

$$\tilde{\sigma}^N = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_N), \quad (7.11)$$

где

$$\tilde{\sigma}_1 \equiv \sigma_1 \in \Sigma_1, \quad \tilde{\sigma}_{i+1}\colon \{z^i | F(z^i) \neq \varnothing\} \to \Sigma_{i+1}, \quad i \geqslant 1, \quad (7.12)$$

а Σ_i есть множество всех вероятностных мер на некоторой σ-алгебре пространства элементарных событий X_i, $i \geqslant 1$ (конечно, можно было бы, как и в случае детерминированных алгоритмов (3.1), при каждом $i \geqslant 1$ задавать отображение $\tilde{\sigma}_{i+1}$ на всем множестве $X_1 \times \ldots \times X_i \times \times Y_1 \times \ldots \times Y_i$). Множество всех алгоритмов $\tilde{\sigma}^N$ вида (7.11), (7.12) обозначим через $\tilde{\Sigma}^N$.

Применение стохастического последовательного алгоритма $\tilde{\sigma}^N$ к некоторой функции $f \in F$ состоит в выборе случайным образом, в соответствии с вероятностным распределением σ_1, некоторого $x_1 \in X_1$, вычислении $y_1 = = x_1(f)$, определении $\sigma_2 = \tilde{\sigma}_2(z^1)$, случайном выборе $x_2 \in \in X_2$ в соответствии с σ_2 и т. д.

Результат, гарантированный алгоритмом $\tilde{\sigma}^N$ на классе F, равен величине

$$\sup_{f \in F} \int_{X_1} \tilde{\sigma}_1 \{dx_1\} \int_{X_2} \tilde{\sigma}_2 (x_1, x_1(f)) \{dx_2\} \ldots$$
$$\ldots \int_{X_N} \tilde{\sigma}_N (x^{N-1}, x^{N-1}(f)) \{dx_N\} \varepsilon (x^N, f),$$

где $x^{N-1}(f) = (x_1(f), \ldots, x_{N-1}(f))$, а критерий $\varepsilon(x^N, f)$ определен формулой (4.14).

Алгоритм $\tilde{\sigma}_0^N = (\tilde{\sigma}_1^0, \ldots, \tilde{\sigma}_N^0) \in \tilde{\Sigma}^N$ называется *оптимальным по точности в множестве стохастических последовательных алгоритмов* $\tilde{\Sigma}^N$ на классе F, если

$$\sup_{f \in F} \int_{X_1} \tilde{\sigma}_1^0 \{dx_1\} \int_{X_2} \tilde{\sigma}_2^0 (x_1, x_1(f)) \{dx_2\} \ldots$$
$$\ldots \int_{X_N} \tilde{\sigma}_N^0 (x^{N-1}, x^{N-1}(f)) \{dx^N\} \varepsilon (x^N, f) =$$
$$= \min_{\tilde{\sigma}^N \in \tilde{\Sigma}^N} \sup_{f \in F} \int_{X_1} \tilde{\sigma}_1 \{dx_1\} \int_{X_2} \tilde{\sigma}_2 (x_1, x_1(f)) \{dx_2\} \ldots$$
$$\ldots \int_{X_N} \tilde{\sigma}_N (x^{N-1}, x^{N-1}(f)) \{dx_N\} \varepsilon (x^N, f).$$

По аналогии с теоремой 4.6 можно показать, что для наилучшего гарантированного при применении алгорит-

мов из $\widetilde{\Sigma}^N$ на классе F результата справедливо равенство

$$\inf_{\widetilde{\sigma}^N \in \widetilde{\Sigma}^N} \sup_{f \in F} \int_{X_1} \widetilde{\sigma}_1 \{dx_1\} \int_{X_2} \widetilde{\sigma}_2 (x_1, x_1(f)) \{dx_2\} \ldots$$

$$\ldots \int_{X_N} \widetilde{\sigma}_N (x^{N-1}, x^{N-1}(f)) \{dx_N\} \varepsilon (x^N, f) =$$

$$= \inf_{\sigma_1 \in \Sigma_1} \sup_{f_1 \in F} \int_{X_1} \sigma_1 \{dx_1\} \inf_{\sigma_2 \in \Sigma_2} \sup_{f_2 \in F(x_1, x_1(f_1))} \int_{X_2} \sigma_2 \{dx_2\} \ldots$$

$$\ldots \inf_{\sigma_N \in \Sigma_N} \sup_{f_N \in F(x^{N-1}, x^{N-1}(f_{N-1}))} \int_{X_N} \sigma_N \{dx_N\} \varepsilon (x^N, f_N). \quad (7.13)$$

В случае, когда допустимы произвольные итоговые операции и вычислитель может рандомизировать выбор β, стохастический последовательный алгоритм задается парой $(\widetilde{\sigma}^N, \widetilde{\tau}) \in \widetilde{\Sigma}^N \times \widetilde{T}$ (см. (7.9)), а гарантированный этим алгоритмом результат равен величине

$$\sup_{f \in F} \int_{X_1} \widetilde{\sigma}_1 \{dx_1\} \int_{X_2} \widetilde{\sigma}_2 (x_1, x_1(f)) \{dx_2\} \ldots$$

$$\ldots \int_{X_N} \widetilde{\sigma}_N (x^{N-1}, x^{N-1}(f)) \{dx_N\} \int_B \widetilde{\tau} (x^N, x^N(f)) \{d\beta\} \gamma (S(f), \beta).$$

Для этого случая по аналогии с уже введенными понятиями оптимальности можно ввести определение оптимального алгоритма и получить выражение для наилучшего гарантированного при применении алгоритмов из $\widetilde{\Sigma}^N \times \widetilde{T}$ на классе F результата.

Сделаем одно важное замечание к определению стохастического последовательного алгоритма. В принятом нами определении выбор возможных альтернатив рандомизируется вычислителем локально, т. е. выбор одной из возможных альтернатив производится случайным образом в каждой реализуемой ситуации (в терминах теории игр — позиции) z^i. Такие алгоритмы в теории игр называются *стратегиями поведения* [149]. Можно предложить другой — глобальный способ, когда рандомизируется выбор последовательных алгоритмов из множества $\widetilde{A}^N$ (или $\widetilde{X}^N$ — в случае, если итоговая операция фиксирована).

Как известно из теории игр [149], локальный и глобальный способы рандомизации эквивалентны между собой для игры с полной памятью и конечным множеством

альтернатив у каждого игрока в каждой позиции. Эквивалентность понимается в том смысле, что для любой пары стратегий игроков, построенных по одному из способов, существует пара стратегий, построенных по другому способу, позволяющая достичь тех же средних выигрышей, что и при первом способе.

Естественно ожидать, что в нашей модели вычислений, несмотря на бесконечность множества альтернатив в каждой ситуации, локальный и глобальный способы рандомизации при выполнении надлежащих технических предположений также эквивалентны, поскольку соответствующая игра является игрой с полной памятью. Такая эквивалентность доказана применительно к задаче поиска экстремума в [186, гл. 1, теорема 3.4].

4. Оптимальные на один шаг стохастические алгоритмы. Сформулируем понятие оптимального на один шаг стохастического алгоритма, предполагая по-прежнему, что все встречающиеся в данном параграфе операции интегрирования осуществимы. Пусть итоговая операция фиксирована, а алгоритм

$$\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_N, \ldots) \tag{7.14}$$

является последовательностью отображений вида (7.12). Множество всех таких алгоритмов обозначим через $\tilde{\Sigma}$. Их эффективность будем оценивать по критерию, являющемуся осреднением критерия (4.14). При применении алгоритма (7.14) остановка, как и в случае детерминированных алгоритмов, производится или после заранее заданного числа шагов N, или после гарантированного достижения заданной точности ε (см. (4.20)).

Алгоритм $\tilde{\sigma}$ называется *оптимальным на один шаг* в $\tilde{\Sigma}$ на классе F, если

$$\sup_{f \in F} \int_{X_1} \varepsilon(x, f)\, \sigma_1\{dx\} = \min_{\sigma \in \Sigma_1} \sup_{f \in F} \int_{X_1} \varepsilon(x, f)\, \sigma\{dx\}, \tag{7.15}$$

где $\sigma_1 = \tilde{\sigma}_1$; и для любой реализуемой ситуации $z^i = (x^i, y^i)$, $i \geqslant 1$,

$$\sup_{f \in F(z^i)} \int_{X_{i+1}} \varepsilon(x^i, x, f)\, \sigma_{i+1}\{dx\} = \min_{\sigma \in \Sigma_{i+1}} \sup_{f \in F(z^i)} \int_{X_{i+1}} \varepsilon(x^i, x, f)\, \sigma\{dx\}, \tag{7.16}$$

где $\sigma_{i+1} = \tilde{\sigma}_{i+1}(z^i)$.

Заметим, что было бы неправильным определять оптимальный на один шаг стохастический алгоритм, поменяв местами в равенствах (7.15) и (7.16) операции взятия верхней грани и интегрирования. Такой подход означал бы приписывание «природе», которая «выбирает» функцию f, свойств разумного противника (именно, способность «узнать» конкретную реализацию случайной величины x_{i+1}) и немедленно привел бы к выводу об оптимальности на один шаг среди стохастических алгоритмов, оптимального на один шаг детерминированного алгоритма.

Отметим еще, что наихудшая апостериорная точность после $(i+1)$-го шага

$$\sup_{y_{i+1}\in\{y_{i+1}\mid F(z^{i+1})\neq\varnothing\}}\ \sup_{f\in F(z^{i+1})}\varepsilon\left(x^{i+1},f\right)=\sup_{f\in F(z^{i})}\varepsilon\left(x^{i+1},f\right)$$

при любом $\sigma_{i+1}\in\Sigma_{i+1}$ в среднем не лучше гарантированной в среднем априорной перед $(i+1)$-м шагом точности (последняя выражается общим значением левой и правой частей (7.16)), ибо

$$\sup_{f\in F(z^{i})}\int\limits_{X_{i+1}}\varepsilon\left(x^{i+1},f\right)\sigma_{i+1}\{dx_{i+1}\}\leqslant$$

$$\leqslant\int\limits_{X_{i+1}}\sup_{f\in F(z^{i})}\varepsilon\left(x^{i+1},f\right)\sigma_{i+1}\{dx_{i+1}\}.$$

5. Стохастически последовательно-оптимальные алгоритмы. Определим, наконец, понятие стохастически последовательно-оптимального по точности алгоритма, предполагая итоговую операцию $\widetilde{\beta}$ фиксированной, а все фигурирующие ниже операции интегрирования — осуществимыми. Приводимое ниже определение аналогично определению 2 из § 6. Можно было бы дать определение, аналогичное определению 1 из § 6, и, пользуясь представлением (7.13) и соответствующими представлениями для наилучших гарантированных результатов на классах $F(z^i)$, доказать эквивалентность двух определений стохастически последовательно-оптимального по точности алгоритма.

Алгоритм $\widetilde{\sigma}_0^N=(\widetilde{\sigma}_1^0,\ldots,\widetilde{\sigma}_N^0)\in\widetilde{\Sigma}^N$ называется *стохастически последовательно-оптимальным по точности* на

классе F, если внешняя нижняя грань в выражении

$$\inf_{\sigma_1\in\Sigma_1}\sup_{f_1\in F}\int_{X_1}\sigma_1\{dx_1\}\inf_{\sigma_2\in\Sigma_2}\sup_{f_2\in F(x_1,x_1(f_1))}\int_{X_2}\sigma_2\{dx_2\}\ldots$$

$$\ldots\inf_{\sigma_N\in\Sigma_N}\sup_{f_N\in F(x^{N-1},x^{N-1}(f_{N-1}))}\int_{X_N}\sigma_N\{dx_N\}\,\varepsilon(x^N,f_N)$$

для наилучшего гарантированного на классе F результата достигается при $\sigma_1=\tilde{\sigma}_1^0$, а внешняя нижняя грань в выражении

$$\inf_{\sigma_{i+1}\in\Sigma_{i+1}}\sup_{f_{i+1}\in F(z^i)}\int_{X_{i+1}}\sigma_{i+1}\{dx_{i+1}\}\ldots$$

$$\ldots\inf_{\sigma_N\in\Sigma_N}\sup_{f_N\in F(x^{N-1},x^{N-1}(f_{N-1}))}\int_{X_N}\sigma_N\{dx_N\}\,\varepsilon(x^N,f_N)$$

для наилучшего гарантированного на классе $F(z^i)$ результата при любой реализуемой ситуации z^i, $1\leqslant i\leqslant N-1$, достигается при $\sigma_{i+1}=\tilde{\sigma}_{i+1}^0(z^i)$.

ГЛАВА 2

ВЫЧИСЛЕНИЕ ИНТЕГРАЛОВ

В этой главе изучается задача численного интегрирования функций одной и нескольких переменных. Построены оптимальные квадратурные формулы, оптимальные пассивные и последовательные алгоритмы вычисления интегралов, наконец, последовательно-оптимальные алгоритмы для различных функциональных классов. Рассмотрены вопросы учета погрешности вычислений и программной реализации предлагаемых методов.

§ 1. Оптимальные квадратуры для функциональных классов, заданных квазиметриками

Получен ряд общих результатов об оптимальных коэффициентах и оптимальных узлах квадратурных формул для функциональных классов, заданных квазиметриками.

1. Постановка задачи. Пусть K — измеримое множество конечной лебеговой меры в n-мерном евклидовом пространстве $\mathbf{R}^n$, функция $\rho(u, v)$ определена на $K \times K$, суммируема по u при любом фиксированном v и обладает свойствами квазиметрики (2.2) — (2.5) гл. 1. Через F_ρ будем обозначать в гл. 2 класс всех измеримых на K функций, удовлетворяющих неравенству

$$|f(u)-f(v)| \leqslant \rho(u, v), \quad u, v \in K,$$

т. е. дополнительно к предположениям п. 2 § 2 гл. 1 все функции из класса F_ρ предполагаются измеримыми.

Предположим, что информация о функции f, получаемая в процессе вычислений, состоит из N значений функции:

$$x_i(f) = f(x_i), \quad x_i \in K, \quad i = 1, \ldots, N,$$

т. е. в формулах (3.1) гл. 1 можно полагать

$$X_1 = \ldots = X_N = K.$$

Традиционным объектом математических исследований являются методы приближения интегралов $\int\limits_K f(x)\,dx$ суммами вида $\sum\limits_{i=1}^{N} p_i f(x_i)$. Соответствующие формулы приближенного вычисления интегралов называют *квадратурными формулами* или *квадратурами*. Иногда мы будем и саму функцию $\sum\limits_{i=1}^{N} p_i f(x_i)$ также называть квадратурной формулой. Точки $x_1, \ldots, x_N$ называют *узлами*, а числа $p_1, \ldots, p_N$ — *коэффициентами* или *весами* квадратурной формулы.

Оптимальными на классе F коэффициентами квадратурной формулы при фиксированных узлах $x_1, \ldots, x_N$ называют коэффициенты $p_1^0(x^N), \ldots, p_N^0(x^N)$, для которых

$$\sup_{f\in F}\left|\int\limits_K f(x)\,dx - \sum_{i=1}^{N} p_i^0(x^N) f(x_i)\right| = \\ = \min_{p^N\in \mathbf{R}^N}\sup_{f\in F}\left|\int\limits_K f(x)\,dx - \sum_{i=1}^{N} p_i f(x_i)\right|, \quad (1.1)$$

где $x^N = (x_1, \ldots, x_N)$, $p^N = (p_1, \ldots, p_N)$.

Оптимальными на классе F узлами называют узлы $x_1^0, \ldots, x_N^0 \in K$, для которых

$$\min_{p^N\in \mathbf{R}^N}\sup_{f\in F}\left|\int\limits_K f(x)\,dx - \sum_{i=1}^{N} p_i f(x_i^0)\right| = \\ = \min_{x^N\in K^N}\min_{p^N\in \mathbf{R}^N}\sup_{f\in F}\left|\int\limits_K f(x)\,dx - \sum_{i=1}^{N} p_i f(x_i)\right|, \quad (1.2)$$

где K^N обозначает N-ю декартову степень множества K.

Квадратурная формула с оптимальными узлами и коэффициентами называется *оптимальной*. Задача построения оптимальных квадратурных формул интенсивно изучалась во многих работах (см. библиографии в [97, 191, 276]).

Наша цель — для функционального класса $F = F_\rho$ получить оценку эффективности заданной квадратурной формулы, т. е. указать при заданных p^N и x^N способ

вычисления величины

$$\sup_{f\in F}\left|\int_K f(x)\,dx-\sum_{i=1}^{N}p_i f(x_i)\right|;$$

при заданных узлах x^N найти оптимальные коэффициенты $p_1^0(x^N), \ldots, p_N^0(x^N)$ и, наконец, указать способ отыскания оптимальных узлов.

2. Вспомогательные утверждения. Установим некоторые результаты вспомогательного характера, на основе которых будут решены поставленные задачи. До конца данного параграфа будем считать, что $F=F_\rho$. В частности,

$$F(z^i)=\{f\in F_\rho \mid f(x_j)=y_j,\ j=1,\ \ldots,\ i\},$$

где, как и прежде, $z^i=(x^i,\ y^i)$. Для $1\leqslant i\leqslant N$ положим

$$\varphi_{1i}(x)=\max_{j=1,\ldots,i}\{y_j-\rho(x,x_j)\},\varphi_{2i}(x)=\min_{j=1,\ldots,i}\{y_j+\rho(x,x_j)\}, \tag{1.3}$$

$$\Phi_1(z^i)=\int_K \varphi_{1i}(x)\,dx,\qquad \Phi_2(z^i)=\int_K \varphi_{2i}(x)\,dx,$$

$$\Psi_1(z^i,p^i)=\langle p^i,y^i\rangle-\Phi_1(z^i),\ \Psi_2(z^i,p^i)=\Phi_2(z^i)-\langle p^i,y^i\rangle. \tag{1.4}$$

Подчеркнем, что $z^i=(x_1,\ \ldots,\ x_i,\ y_1,\ \ldots,\ y_i)$ не фигурирует явным образом среди аргументов и индексов функций $\varphi_{1i}(x)$, $\varphi_{2i}(x)$ лишь для упрощения обозначений; о зависимости от z^i напоминает индекс i.

Начнем с доказательства трех простых лемм.

Лемма 1.1. *Если* $|y_\mu-y_\nu|\leqslant\rho(x_\mu,\ x_\nu)$ *для всех* $\mu,\ \nu=1,\ \ldots,\ i$, *то* $\varphi_{1i}\in F(z^i)$, $\varphi_{2i}\in F(z^i)$.

Доказательство. Из условия леммы следует, что $y_\nu-\rho(x_\mu,\ x_\nu)\leqslant y_\mu$. Поэтому $\varphi_{1i}(x_\mu)=\max\limits_{\nu=1,\ldots,i}\{y_\nu-\rho(x_\mu,\ x_\nu)\}\leqslant y_\mu$. С другой стороны, $\varphi_{1i}(x_\mu)\geqslant y_\mu-\rho(x_\mu,\ x_\mu)=y_\mu$. Таким образом, $\varphi_{1i}(x_\mu)=y_\mu$, $\mu=1,\ \ldots,\ i$. Так как $\varphi_{1i}\in F$ в силу лемм 2.1 и 2.7 гл. 1, то $\varphi_{1i}\in F(z^i)$. Аналогично доказывается, что $\varphi_{2i}\in F(z^i)$. $\triangle$

Лемма 1.2. *Если* $f\in F(z^i)$, *то*

$$\varphi_{1i}(x)\leqslant f(x)\leqslant\varphi_{2i}(x),\quad x\in K.$$

Доказательство. Так как $f\in F(z^i)$, то

$$-\rho(x,\ x_j)\leqslant f(x)-f(x_j)=f(x)-y_j\leqslant\rho(x,\ x_j),\quad j=1,\ \ldots,\ i,$$

откуда

$$\max_{j=1,\ldots,i} \{y_j - \rho(x, x_j)\} \leqslant f(x) \leqslant \min_{j=1,\ldots,i} \{y_j + \rho(x, x_j)\}. \quad \triangle$$

Леммы 1.1 и 1.2 показывают, что функция φ_{1i} служит *точной минорантой* для функций из $F(z^i)$, а функция φ_{2i} — *точной мажорантой.*

Лемма 1.3. 1) *Множество* $l(x^i) \overset{\text{def}}{=} \{y^i \mid F(z^i) \neq \varnothing\}$ *представимо в виде*

$$l(x^i) = \{y^i \mid |y_\mu - y_\nu| \leqslant \rho(x_\mu, x_\nu), \ \mu, \nu = 1, \ldots, i\}.$$

2) *Множество* $l(x^i)$ *выпукло.*

Доказательство. 1) Включение

$$l(x^i) \subset \{y^i \mid |y_\mu - y_\nu| \leqslant \rho(x_\mu, x_\nu), \ \mu, \nu = 1, \ldots, i\}$$

очевидно. Справедливость обратного включения вытекает из леммы 1.1. Тем самым первое утверждение леммы доказано.

2) Пусть $u^i = (u_1, \ldots, u_i) \in l(x^i)$, $v^i = (v_1, \ldots, v_i) \in l(x^i)$, $0 \leqslant \lambda \leqslant 1$. Тогда при всех $\mu, \nu = 1, \ldots, i$ имеем

$$|[\lambda u_\mu + (1-\lambda) v_\mu] - [\lambda u_\nu + (1-\lambda) v_\nu]| \leqslant$$
$$\leqslant \lambda |u_\mu - u_\nu| + (1-\lambda) |v_\mu - v_\nu| \leqslant \rho(x_\mu, x_\nu),$$

то есть $\lambda u^i + (1-\lambda) v^i \in l(x^i)$. $\triangle$

Пусть m принимает значения 1, 2. Обозначим через $K_{mi}(z^N)$, $i = 1, \ldots, N$, произвольный набор множеств, которые образуют разбиение K, т. е.

$$\bigcup_{i=1}^{N} K_{mi}(z^N) = K, \ K_{mi}(z^N) \cap K_{mj}(z^N) = \varnothing \ \text{при} \ i \neq j,$$

и удовлетворяют условиям

$$K_{mi}(z^N) \quad \text{измеримо}, \quad i = 1, \ldots, N,$$
$$y_i + (-1)^m \rho(x, x_i) = \varphi_{mN}(x) \quad \text{при} \quad x \in K_{mi}(z^N). \tag{1.5}$$

Заметим, что в рассматриваемом случае, когда ρ — квазиметрика, такие разбиения так же, как и меры множеств, образующих эти разбиения, могут быть не единственными. Лебегову меру множества A будем обозначать через $\mu(A)$. Определим множество

$$A_m(z^N) = \{g = (g_1, \ldots, g_N) \mid g_i = \mu(K_{mi}(z^N)), \ i = 1, \ldots, N\},$$
$$m = 1, 2.$$

Каждый элемент g этого множества, как видно из определения, соответствует некоторому разбиению (1.5), причем компоненты вектора g являются лебеговыми мерами множеств $K_{mi}(z^N)$, образующих разбиение (1.5).

Лемма 1.4. *Множества $A_1(z^N)$, $A_2(z^N)$ выпуклы и замкнуты.*

Доказательство. Считая z^N фиксированным, не будем явно указывать в ходе доказательства зависимость от z^N множеств K_{mi}, A_m и множеств, определяемых ниже. Положим для любых попарно различных $i_1, \ldots, i_k \in \{1, \ldots, N\}$ $(1 \leqslant k \leqslant N)$

$$B_{i_1,\ldots,i_k} = \{x \in K \mid y_i - \rho(x, x_i) = \varphi_{1N}(x) > y_j - \rho(x, x_j),$$
$$i \in \{i_1, \ldots, i_k\},\ j \in \{1, \ldots, N\} \setminus \{i_1, \ldots, i_k\}\}.$$

Пусть $\alpha = (\alpha^1, \ldots, \alpha^N) \in A_1$, K_{1i}^{α} — разбиение K, удовлетворяющее условию (1.5) и такое, что $\alpha^i = \mu(K_{1i}^{\alpha})$, $i = 1, \ldots, N$. Положим для $k \geqslant 2$

$$B_{i_1,\ldots,i_k}^{\alpha i} = K_{1i}^{\alpha} \cap B_{i_1,\ldots,i_k}, \quad \alpha_{i_1,\ldots,i_k}^{i} = \mu(B_{i_1,\ldots,i_k}^{\alpha i}),$$
$$i \in \{i_1, \ldots, i_k\}.$$

Пусть $\beta = (\beta^1, \ldots, \beta^N) \in A_1$, $0 \leqslant \lambda \leqslant 1$ и для β используются те же обозначения, что и для α. Пусть $\gamma_{i_1,\ldots,i_k}^{i} = \lambda\alpha_{i_1,\ldots,i_k}^{i} + (1-\lambda)\beta_{i_1,\ldots,i_k}^{i}$, $i \in \{i_1, \ldots, i_k\}$. Мы можем разбить $B_{i_1,\ldots,i_k}$ на множества $B_{i_1,\ldots,i_k}^{\gamma i}$ такие, что $\mu(B_{i_1,\ldots,i_k}^{\gamma i}) = \gamma_{i_1,\ldots,i_k}^{i}$, $i \in \{i_1, \ldots, i_k\}$. Легко видеть теперь, что множества

$$K_{1i}^{\gamma} \stackrel{\text{def}}{=} \bigcup_{\{i_1,\ldots,i_k\} \ni i;\, k \geqslant 2} B_{i_1,\ldots,i_k}^{\gamma i} \cup B_i, \quad i = 1, \ldots, N,$$

удовлетворяют условию (1.5). Полагая $\gamma^i = \mu(K_{1i}^{\gamma})$, $\gamma = (\gamma^1, \ldots, \gamma^N)$, имеем $\gamma \in A_1$. Но при $i = 1, \ldots, N$

$$\gamma^i = \mu(K_{1i}^{\gamma}) = \sum_{\{i_1,\ldots,i_k\} \ni i;\, k \geqslant 2} \gamma_{i_1,\ldots,i_k}^{i} + \mu(B_i) =$$
$$= \lambda\left[\sum_{\{i_1,\ldots,i_k\} \ni i;\, k \geqslant 2} \alpha_{i_1,\ldots,i_k}^{i} + \mu(B_i)\right] +$$
$$+ (1-\lambda)\left[\sum_{\{i_1,\ldots,i_k\} \ni i;\, k \geqslant 2} \beta_{i_1,\ldots,i_k}^{i} + \mu(B_i)\right] = \lambda\alpha^i + (1-\lambda)\beta^i.$$

Таким образом, $\gamma = \lambda\alpha + (1-\lambda)\beta \in A_1$, что доказывает выпуклость A_1.

Докажем замкнутость. Пусть $\alpha(t) = (\alpha^1(t), \ldots, \alpha^N(t)) \in A_1$, $\alpha(t) \to \alpha$ при $t \to \infty$. Определим $\alpha^i_{i_1,\ldots,i_k}(t)$, $i \in \{i_1, \ldots, i_k\}$, $k \geqslant 2$, по аналогии с определением величин $\alpha^i_{i_1,\ldots,i_k}$ в первой части доказательства. Выберем $t_r \to \infty$ таким образом, что $\alpha^i_{i_1,\ldots,i_k}(t_r) \to \alpha^i_{i_1,\ldots,i_k}$ при $r \to \infty$ для всех допустимых $i_1, \ldots, i_k, i, k$. Пусть множества $B^{\alpha i}_{i_1,\ldots,i_k}$ образуют разбиение $B_{i_1,\ldots,i_k}$, причем $\mu(B^{\alpha i}_{i_1,\ldots,i_k}) = \alpha^i_{i_1,\ldots,i_k}$, $i \in \{i_1, \ldots, i_k\}$, $k \geqslant 2$. Тогда разбиение K на множества

$$K^\alpha_{1i} = \bigcup_{\{i_1,\ldots,i_k\} \ni i; k \geqslant 2} B^{\alpha i}_{i_1,\ldots,i_k} \cup B_i, \quad i = 1, \ldots, N,$$

соответствует α, т. е. $\mu(K^\alpha_{1i}) = \alpha^i$, $i = 1, \ldots, N$. Для доказательства заметим, что

$$\mu(K^\alpha_{1i}) = \sum_{\{i_1,\ldots,i_k\} \ni i; k \geqslant 2} \alpha^i_{i_1,\ldots,i_k} + \mu(B_i) =$$
$$= \lim_{r \to \infty} \left[\sum_{\{i_1,\ldots,i_k\} \ni i; k \geqslant 2} \alpha^i_{i_1,\ldots,i_k}(t_r) + \mu(B_i) \right] =$$
$$= \lim_{r \to \infty} \alpha^i(t_r) = \alpha^i, \quad i = 1, \ldots, N.$$

Следовательно, $\alpha \in A_1$, и, стало быть, A_1 замкнуто. Аналогично доказывается выпуклость и замкнутость A_2. $\triangle$

Напомним, что множество

$$\partial_{y^N} \Phi(y^N) =$$
$$= \{g \in \mathbf{R}^N \mid \Phi(y^N + \Delta y) - \Phi(y^N) \geqslant \langle g, \Delta y \rangle$$
$$\text{для любых } \Delta y \in \mathbf{R}^N\}$$

называется субдифференциалом в точке y^N выпуклой функции Φ: $\mathbf{R}^N \to \mathbf{R}$, а множество

$$\partial_{y^N} \Psi(y^N) =$$
$$= \{g \in \mathbf{R}^N \mid \Psi(y^N + \Delta y) - \Psi(y^N) \leqslant \langle g, \Delta y \rangle$$
$$\text{для любых } \Delta y \in \mathbf{R}^N\}$$

— **субдифференциалом в точке y^N вогнутой функции** Ψ: $\mathbf{R}^N \to \mathbf{R}$.

Лемма 1.5. *Функция* $\Phi_1(z^N)$ *выпукла по* y^N *на* $\mathbf{R}^N$, *функция* $\Phi_2(z^N)$ *вогнута по* y^N *на* $\mathbf{R}^N$. *При этом*

$$\partial_{y^N}\Phi_1(z^N) = A_1(z^N), \quad \partial_{y^N}\Phi_2(z^N) = A_2(z^N).$$

Доказательство. Выпуклость Φ_1 очевидна. Для любых $K_{1i}(z^N)$, $i=1, \ldots, N$, образующих разбиение множества K вида (1.5), и для $\Delta y = (\Delta y_1, \ldots, \Delta y_N)$ получаем:

$$\Phi_1(x^N, y^N + \Delta y) - \Phi_1(x^N, y^N) =$$

$$= \int_K \Big[\max_{j=1,\ldots,N} \{y_j + \Delta y_j - \rho(x, x_j)\} -$$

$$- \max_{j=1,\ldots,n} \{y_j - \rho(x, x_j)\}\Big] dx =$$

$$= \sum_{i=1}^N \int_{K_{1i}(z^N)} \Big[\max_{j=1,\ldots,N} \{y_j + \Delta y_j - \rho(x, x_j)\} -$$

$$- y_i + \rho(x, x_i)\Big] dx \geqslant \sum_{i=1}^N \int_{K_{1i}(z^N)} \Delta y_i \, dx = \sum_{i=1}^N \Delta y_i \mu\,(K_{1i}(z^N)),$$

откуда

$$\partial_{y^N}\Phi_1(z^N) \supset A_1(z^N). \tag{1.6}$$

Пусть

$$K_{11}^0(z^N) = \{x \mid y_1 - \rho(x, x_1) = \varphi_{1N}(x)\},$$

$$K_{1i}^0(z^N) = \{x \mid y_i - \rho(x, x_i) = \varphi_{1N}(x)\} \setminus \bigcup_{j=1}^{i-1} K_{1j}^0(z^N),$$

$$i = 2, \ldots, N.$$

Ясно, что множества $K_{1i}^0(z^N)$, $i=1, \ldots, N$, образуют при любом y^N разбиение множества K, удовлетворяющее условию (1.5), и

$$\mu\,(K_{1i}^0(x^N, y^N + \delta\Delta y)) \to \mu\,(K_{1i}^{\Delta y}(z^N)) \quad \text{при} \quad \delta \to 0+$$

для любого $\Delta y \in \mathbf{R}^N$. При этом множества $K_{1i}^{\Delta y}(z^N)$, $i = 1, \ldots, N$, также образуют разбиение, удовлетворяющее условию (1.5). Таким образом, для любых $\Delta y \in \mathbf{R}^N$, $\delta > 0$

$$\sup_{g \in \partial_{y^N}\Phi_1(z^N)} \langle g, \delta\Delta y\rangle \leqslant \Phi_1(x^N, y^N + \delta\Delta y) - \Phi_1(x^N, y^N) =$$

$$= \int_K \Big[\max_{j=1,\ldots,N} \{y_j + \delta\Delta y_j - \rho(x, x_j)\} - \max_{j=1,\ldots,N} \{y_j - \rho(x, x_j)\}\Big] dx =$$

$$= \sum_{i=1}^{N} \int\limits_{K_{1i}^{0}(x^N, y^N+\delta\Delta y)} \Big[y_i + \delta\Delta y_i - \rho(x, x_i) -$$

$$- \max_{j=1,\ldots,N} \{y_j - \rho(x, x_j)\} \Big] dx \leqslant \delta \sum_{i=1}^{N} \Delta y_i \mu\big(K_{1i}^{0}(x^N, y^N + \delta\Delta y)\big) =$$

$$= \delta \sum_{i=1}^{N} \Delta y_i \mu\big(K_{1i}^{\Delta y}(z^N)\big) +$$

$$+ \delta \sum_{i=1}^{N} \Delta y_i \big[\mu\big(K_{1i}^{0}(x^N, y^N + \delta\Delta y)\big) - \mu\big(K_{1i}^{\Delta y}(z^N)\big)\big] =$$

$$= \delta \sum_{i=1}^{N} \Delta y_i \mu\big(K_{1i}^{\Delta y}(z^N)\big) + o(\delta).$$

Следовательно, для любого $\Delta y \in \mathbf{R}^N$

$$\sup_{g \in \partial_{y^N}\Phi_1(z^N)} \langle g, \Delta y\rangle \leqslant \langle g^{\Delta y}, \Delta y\rangle + o(\delta)/\delta,$$

где $g^{\Delta y} \overset{\text{def}}{=} \big(\mu\big(K_{11}^{\Delta y}(z^N)\big), \ldots, \mu\big(K_{1N}^{\Delta y}(z^N)\big)\big) \in A_1(z^N)$, и поэтому

$$\sup_{g \in \partial_{y^N}\Phi_1(z^N)} \langle g, \Delta y\rangle \leqslant \sup_{g \in A_1(z^N)} \langle g, \Delta y\rangle.$$

Поскольку, в силу (1.6), справедливо противоположное неравенство, имеем для любого $\Delta y \in \mathbf{R}^N$

$$\sup_{g \in \partial_{y^N}\Phi_1(z^N)} \langle g, \Delta y\rangle = \sup_{g \in A_1(z^N)} \langle g, \Delta y\rangle.$$

Отсюда, согласно [213, лемма 1.1], следует, что $\partial_{y^N}\Phi_1(z^N) = A_1(z^N)$, так как $\partial_{y^N}\Phi_1(z^N)$ в силу [213, теорема 1.1], а $A_1(z^N)$ в силу леммы 1.4 — выпуклые замкнутые множества. Аналогично доказывается равенство $\partial_{y^N}\Phi_2(z^N) = A_2(z^N)$. △

3. Об оценке эффективности заданной квадратурной формулы. Перейдем теперь непосредственно к первой из поставленных выше задач — к задаче оценки эффективности квадратурной формулы с заданными узлами x^N и коэффициентами p^N. Используя леммы 1.1 и 1.2 и

обозначения (1.3), (1.4), получаем

$$\max_{f\in F}\left|\int_K f(x)\,dx-\sum_{i=1}^{N}p_i f(x_i)\right|=$$

$$=\max_{y^N\in l(x^N)}\max_{f\in F(z^N)}\left|\int_K f(x)\,dx-\langle p^N,\ y^N\rangle\right|=$$

$$=\max_{y^N\in l(x^N)}\max\left\{\langle p^N,y^N\rangle-\int_K\varphi_{1N}(x)\,dx,\int_K\varphi_{2N}(x)\,dx-\langle p^N,y^N\rangle\right\}=$$

$$=\max\left\{\max_{y^N\in l(x^N)}\Psi_1(z^N,\ p^N),\ \max_{y^N\in l(x^N)}\Psi_2(z^N,\ p^N)\right\}. \quad (1.7)$$

Таким образом, задача сведена к решению двух задач максимизации вогнутых функций на выпуклом в силу леммы 1.3 множестве $l(x^N)$, т. е. к решению двух задач выпуклого программирования. Следующая теорема дает необходимые и достаточные условия экстремума для этих задач.

Теорема 1.1 [251]. *Для того чтобы вектор* $y^N(m)=(y_1(m),\ \ldots,\ y_N(m))$ *реализовывал максимум функции* $\Psi_m(x^N,\ y^N,\ p^N)$ *на множестве* $l(x^N)$ *при фиксированных* x^N, p^N, *необходимо и достаточно, чтобы существовали множества* $K_{mi}(x^N,\ y^N(m))$, $i=1,\ \ldots,\ N$, *удовлетворяющие условиям* (1.5) *при* $y^N=y^N(m)$ *и числа* λ_m^{jk}, $j,\ k=1,\ \ldots,\ N$, *такие, что*

$$(-1)^m\left[\mu\left(K_{mi}(x^N,\ y^N(m))\right)-p_i\right]=\sum_{j,k=1}^{N}\lambda_m^{jk}g_i^{jk},$$

$$\lambda_m^{jk}(y_j(m)-y_k(m)-\rho(x_j,\ x_k))=0,\quad i,\ j,\ k=1,\ \ldots,\ N,$$

где

$$g_i^{jk}=\begin{cases}1, & i=j\neq k,\\ -1, & i=k\neq j,\\ 0 & \text{в других случаях.}\end{cases}$$

Доказательство. Напомним, что $\Psi(Y)=\max\limits_{Y'\in A}\Psi(Y')$ для выпуклого множества A и вогнутой функции Ψ, если и только если

$$\Gamma(Y)^*\cap[-\partial_Y\Psi(Y)]\neq\varnothing, \quad (1.8)$$

где $\Gamma(Y)^*$ — конус, двойственный к конусу $\Gamma(Y)$ допустимых в точке Y относительно множества A направлений [213].

Опишем конусы $\Gamma(y^N)$, $\Gamma(y^N)^*$ при $A = l(x^N)$. По лемме 1.3

$$l(x^N) = \{y^N = (y_1, \ldots, y_N) \mid y_j - y_k \leqslant \rho(x_j, x_k), \quad j, k = 1, \ldots, N\},$$

поэтому

$$\Gamma(y^N) = \{\Delta y = (\Delta y_1, \ldots, \Delta y_N) \mid \Delta y_j - \Delta y_k \leqslant 0, \quad \text{если} \quad y_j - y_k = \rho(x_j, x_k)\}.$$

Применяя лемму Фаркаша, имеем

$$\Gamma(y^N)^* = \left\{-\sum_{j,k=1}^{N} \lambda^{jk} g^{jk} \,\middle|\, \lambda^{jk}(y_j - y_k - \rho(x_j, x_k)) = 0, \quad \lambda^{jk} \geqslant 0, \quad j, k = 1, \ldots, N\right\}, \tag{1.9}$$

где $g^{jk} = (g_1^{jk}, \ldots, g_N^{jk})$.

Поскольку функция Φ_1 выпукла, а Φ_2 вогнута по y^N, то Ψ_1, Ψ_2 — вогнутые по y^N функции, и по лемме 1.5

$$\partial_{y^N} \Psi_m(z^N, p^N) = \{g = (g_1, \ldots, g_N) \mid g_i = (-1)^m [\mu(K_{mi}(z^N)) - p_i], \quad i = 1, \ldots, N\}, \quad m = 1, 2. \tag{1.10}$$

Утверждение теоремы вытекает непосредственно из (1.8)—(1.10). △

Условия оптимальности, содержащиеся в теореме 1.1, полезны при рассмотрении различных вопросов, связанных с оценкой эффективности квадратурных формул, в частности, вопросов построения оптимальных квадратурных формул, в том числе при заданных ограничениях на коэффициенты. В то же время эти условия, как и условия оптимальности в иных задачах, не приводят непосредственно к численному методу решения. Далее теорема 1.1 не используется.

Доказательство теоремы 1.2, отвечающей на вопрос об оптимальных коэффициентах квадратур, будет получено с помощью необходимого и достаточного условия максимума в задаче безусловной оптимизации для вогнутой функции Ψ, т. е. задаче отыскания $\Psi(Y) = \max_{Y' \in \mathbf{R}^N} \Psi(Y')$. Это условие, являющееся частным случаем (1.8) при $A = \mathbf{R}^N$, имеет вид

$$0 \in \partial_Y \Psi(Y). \tag{1.11}$$

4. Об отыскании оптимальных коэффициентов и оптимальных узлов квадратурных формул. Следующая теорема дает ответ на вопрос об оптимальных коэффициентах $p_1^0(x^N), \ldots, p_N^0(x^N)$ при заданных узлах x^N квадратурной формулы, а следствие из нее позволяет в ряде случаев решить задачу об оптимальных узлах. При более ограничительных предположениях некоторые близкие результаты получены в работах [7, 8, 129, 164, 314].

Пусть $\theta^N = (\theta, \ldots, \theta)$, где $\theta \in \mathbf{R}$ — произвольное число. Нетрудно видеть, что множество всех разбиений K вида $K_{mi}(x^N, \theta^N)$, $i = 1, \ldots, N$, не зависит ни от θ, ни от m. Обозначим такое произвольное разбиение через $K_i(x^N)$, $i = 1, \ldots, N$. Иными словами, $K_1(x^N), \ldots, K_N(x^N)$ — произвольный набор измеримых, попарно не пересекающихся множеств, объединение которых есть K, и таких, что

$$\rho(x, x_i) = \min_{j=1,\ldots,N} \rho(x, x_j) \text{ при } x \in K_i(x^N), \quad i = 1, \ldots, N.$$

Назовем такие разбиения ρ-*разбиениями Вороного — Дирихле*. В рассматриваемом случае, когда ρ — квазиметрика, множества $K_i(x^N)$, образующие разбиения Вороного — Дирихле, и их меры $\mu(K_i(x^N))$ определяются, вообще говоря, неоднозначно.

Теорема 1.2 [251]. *Квадратурная формула с коэффициентами* $p_i^0(x^N) \overset{\text{def}}{=} \mu(K_i(x^N))$, $i = 1, \ldots, N$, *оптимальна на классе* F_ρ *среди квадратурных формул с фиксированными узлами* $x_1, \ldots, x_N$, *а гарантированный ею результат (максимальная погрешность) равен*

$$\int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx.$$

Отметим, что в силу неоднозначности определения мер $\mu(K_i(x^N))$ оптимальные коэффициенты $p_i^0(x^N)$ определяются также, вообще говоря, неоднозначно.

Доказательство. Для любого $p^N \in \mathbf{R}^N$ имеем

$$\max_{f \in F} \left| \int_K f(x)\, dx - \sum_{i=1}^N p_i f(x_i) \right| \geqslant \max_{y^N \in l(x^N)} \Psi_1(z^N, p^N) \geqslant$$

$$\geqslant \Psi_1(x^N, \theta^N, p^N) = \theta \sum_{i=1}^N p_i - \int_K \max_{i=1,\ldots,N} \{\theta - \rho(x, x_i)\}\, dx =$$

$$= \theta\left(\sum_{i=1}^{N} p_i - \mu(K)\right) + \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\,dx \geqslant$$
$$\geqslant \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\,dx, \quad (1.12)$$

где первое неравенство справедливо в силу (1.7), второе неравенство справедливо, так как $\theta^N \in l(x^N)$, наконец, третье неравенство справедливо, поскольку $\theta \in \mathbf{R}$ может быть выбрано так, что $\theta\left(\sum_{i=1}^{N} p_i - \mu(K)\right) \geqslant 0$.

При $y^N = \theta^N$, $p^N = p_0^N(x^N) \overset{\text{def}}{=} (p_1^0(x^N), \ldots, p_N^0(x^N))$ формула (1.10) принимает вид

$$\partial_{y^N}\Psi_m(x^N, \theta^N, p_0^N(x^N)) =$$
$$= \{g = (g_1, \ldots, g_N) \mid g_i = (-1)^m(\mu(K_i(x^N)) - p_i^0(x^N)),$$
$$i = 1, \ldots, N\}, \quad m = 1, 2.$$

Каждый элемент g данного множества соответствует некоторому разбиению Вороного — Дирихле. В силу определения оптимальных коэффициентов для произвольного набора оптимальных коэффициентов $p_0^N(x^N)$ условие оптимальности (1.11) выполнено для функций Ψ_m при $y^N = \theta^N$, $p^N = p_0^N(x^N)$, т. е.

$$0 \in \partial_{y^N}\Psi_m(x^N, \theta^N, p_0^N(x^N)), \quad m = 1, 2.$$

Поэтому, учитывая, что $\theta^N \in l(x^N)$, имеем

$$\Psi_m(x^N, \theta^N, p_0^N(x^N)) = \max_{y^N \in \mathbf{R}^N} \Psi_m(x^N, y^N, p_0^N(x^N)) =$$
$$= \max_{y^N \in l(x^N)} \Psi_m(x^N, y^N, p_0^N(x^N)), \quad m = 1, 2. \quad (1.13)$$

Поскольку

$$\sum_{i=1}^{N} p_i^0(x^N) = \sum_{i=1}^{N} \mu(K_i(x^N)) = \mu(K),$$

получаем при $p^N = p_0^N(x^N)$ из равенств, содержащихся в цепочке (1.12), что

$$\Psi_1(x^N, \theta^N, p_0^N(x^N)) = \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\,dx.$$

Аналогично,

$$\Psi_2(x^N, \theta^N, p_0^N(x^N)) = \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx.$$

Следовательно, учитывая (1.7) и (1.13), имеем

$$\max_{f\in F} \left| \int_K f(x)\, dx - \sum_{i=1}^N p_i^0(x^N) f(x_i) \right| = \\ = \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx. \quad (1.14)$$

Утверждение теоремы вытекает непосредственно из (1.12) и (1.14). △

Доказанная теорема дает также способ отыскания оптимальных узлов. Сформулируем его в виде следствия из теоремы.

Следствие. *Решение задачи отыскания узлов оптимальной квадратурной формулы сводится к отысканию в K таких точек $x_1^0, \ldots, x_N^0$, что*

$$\int_K \min_{i=1,\ldots,N} \rho(x, x_i^0)\, dx = \min_{x^N\in K^N} \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx. \quad (1.15)$$

Наилучший на классе F_ρ в множестве всех квадратурных формул результат равен

$$m_N \overset{\text{def}}{=} \inf_{x^N\in K^N} \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx. \ \triangle$$

Теорема 1.3. *Квадратурная формула с коэффициентами $p_i^0(x^N)$, $i=1, \ldots, N$, является оптимальной итоговой операцией при фиксированных $x_1, \ldots, x_N \in K$. Оптимальная квадратурная формула, т. е. квадратурная формула с узлами $x_0^N = (x_1^0, \ldots, x_N^0)$, удовлетворяющими условию (1.15), и коэффициентами $p_i^0(x_0^N)$, $i=1, \ldots, N$, является оптимальным по точности на классе F_ρ алгоритмом в множестве всех пассивных алгоритмов A^N, при этом наилучший гарантированный результат равен m_N.*

Доказательство. Для произвольных фиксированных $x_1, \ldots, x_N \in K$

$$\min_{\tilde\beta\in\tilde B_N} \sup_{f\in F} \left| \int_K f(x)\, dx - \tilde\beta(x_1, \ldots, x_N, f(x_1), \ldots, f(x_N)) \right| = \\ = \min_{\theta\in\Theta} \sup_{f\in F} \left| \int_K f(x)\, dx - \theta(f(x_1), \ldots, f(x_N)) \right| =$$

$$= \sup_{f \in F} \left| \int_K f(x)\,dx - \sum_{i=1}^{N} p_i^0(x^N) f(x_i) \right| = \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\,dx, \tag{1.16}$$

где $\widetilde{B}_N$ — множество всевозможных итоговых операций, Θ — множество всех числовых функций θ переменных $(y_1, \ldots, y_N) \in \{(y_1, \ldots, y_N) \mid y_1 = f(x_1), \ldots, y_N = f(x_N), f \in F\}$.

В левой и правой частях первого равенства (1.16) стоит одна и та же величина, переписанная в разных обозначениях. Справедливость второго равенства показывают лемма 2.8 и теорема 4.2 гл. 1, а также теорема 1.2: в силу леммы 2.8 гл. 1 функциональный класс $F = F_\rho$ является выпуклым центрально-симметричным множеством с центром 0; поэтому, по теореме 4.2 гл. 1, среди оптимальных итоговых операций при фиксированных $x_1, \ldots, x_N$ существует квадратурная формула; наконец, в силу теоремы 1.2 этой квадратурной формулой является квадратура с коэффициентами $p_i^0(x^N)$, $i = 1, \ldots, N$. Справедливость третьего равенства также следует из теоремы 1.2. Второе равенство в (1.16) — это и есть первое из доказываемых утверждений. Для доказательства второго утверждения достаточно проминимизировать по $x^N = (x_1, \ldots, x_N) \in K^N$ все соединенные в (1.16) знаками равенства величины. $\triangle$

Задача (1.15) близка к задачам дискретной геометрии. Так, если ρ — евклидова метрика, то задача (1.15) тесно связана с задачей оптимального покрытия множествами K шарами равного радиуса и с задачей наименее плотного покрытия n-мерного пространства единичными шарами. Этот вопрос будет подробно рассмотрен в следующем параграфе.

Найти в явном виде решение задачи (1.15) удается сравнительно редко (см. § 2). При необходимости многократного использования пассивного алгоритма с заданным числом узлов N может оказаться целесообразным решить задачу (1.15) численно.

При небольших значениях N и размерности пространства n удается приближенно найти решение, например, при помощи следующего итерационного процесса. Пусть

$$\eta(x^N) \stackrel{\text{def}}{=} \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\,dx = \sum_{i=1}^{N} \int_{K_i(x^N)} \rho(x, x_i)\,dx,$$

$$D_i(x^N) \stackrel{\text{def}}{=} \int\limits_{K_i(x^N)} \rho(x, x_i)\, dx.$$

На $(k+1)$-й итерации функция η минимизируется по векторной переменной $x_{m(k)}$ на отрезке $[x_{m(k)}(k), a(k)]$, где вектор $x^N(k) = (x_1(k), \ldots, x_N(k))$ получен в результате k-й итерации, номер $m(k) \in \{1, \ldots, N\}$ определен соотношением

$$D_{m(k)}(x^N(k)) = \max_{i=1,\ldots,N} D_i(x^N(k)),$$

наконец,

$$a(k) = \arg \max_{x \in K_{m(k)}(x^N(k))} \rho(x, x_{m(k)}(k)).$$

В случае, когда $n=2$, K — единичный квадрат, ρ — евклидова метрика, описанный алгоритм дает следующие приближенные значения для наилучшей гарантированной точности: $m_2 \approx 0.297$ (на самом деле $m_2 = \frac{1}{6}\left(\frac{\sqrt{5}}{4} + \ln\left(\frac{\sqrt{5}+1}{2}\right)\right) + \frac{1}{24}\left(\sqrt{5} - \frac{1}{2}\ln\left(\frac{1}{\sqrt{5}+2}\right)\right) = 0.2966\ldots$), $m_3 \approx 0.237$, $m_4 \approx 0.187$, $m_5 \approx 0.175$.

Теорема 1.2 позволяет также производить по критерию $\eta(x^N)$ численное сравнение эффективности для задачи интегрирования различных способов «равномерного» в том или ином смысле расположения точек $x_1, \ldots, x_N$ в множестве K.

Сравним, например, три таких способа, считая попрежнему, что $n=2$, K — единичный квадрат, ρ — евклидова метрика.

Первый способ — выбор точек из решетки Γ_1^n (см. замечание 6 к теореме 2.3 следующего параграфа). При $n=2$ основной репер этой решетки задается правильным треугольником. Длина его стороны выбирается так, чтобы в единичный квадрат попало не менее N точек решетки и чтобы при большей длине стороны в единичный квадрат попадало менее N точек.

Второй способ — выбор точек из «кубической» решетки, построенной в [301]. При $N = m^n$ (m — целое) эта решетка совпадает с обычной кубической решеткой.

Наконец, третий способ — выбор точек из ЛП$_\tau$-последовательности, построенной согласно [231, гл. 6, § 3].

Результаты вычислений представим в табл. 1, элементами которой являются приближенно вычисленные значения величины $\eta(x^N) = \int\limits_K \min\limits_{i=1,\ldots,N} \rho(x, x_i)dx$. Таблица показывает, что, начиная с $N=7$, наиболее эффективными являются квадратурные формулы, построенные с помощью решетки Γ_1^n. Для сравнения укажем, что при

Таблица 1

N	5	7	11	17
Γ_1^n	0.227	0.151	0.127	0.103
«Кубическая» решетка	0.193	0.157	0.130	0.104
$ЛП_\tau$	0.241	0.179	0.142	0.115

$N=7$, например, осреднение по 50 испытаниям величины $\eta(x^N)$ в случае, если $x_1, \ldots, x_N$ — независимые, равномерно распределенные на K случайные точки, дает результат 0.211, при $N=17$ — 0.128.

5. Центральная итоговая операция в задаче интегрирования. Получим выражение для наилучшей гарантированной после завершения информационных вычислений точности $\varepsilon_N(z^N)$ и установим вид центральной итоговой операции $\tilde{\beta}_*$ (см. (4.3), (4.4) гл. 1). При этом по-прежнему считаем, что $F = F_\rho$. Кроме того, $S(f) = \int\limits_K f(x)\,dx$ и $B = \mathbf{R}$ (см. п. 1 § 1 гл. 1), т. е. расстояние между элементами $S(f_1)$, $S(f_2)$ есть $|S(f_1) - S(f_2)|$.

Теорема 1.4. *Функция*

$$\tilde{\beta}_*(z^N) = \int\limits_K \frac{1}{2}[\varphi_{1N}(x) + \varphi_{2N}(x)]\,dx$$

задает центральную итоговую операцию, при этом

$$\varepsilon_N(z^N) = \int\limits_K \frac{1}{2}[\varphi_{2N}(x) - \varphi_{1N}(x)]\,dx.$$

Доказательство. В силу лемм 1.1 и 1.2 заданные формулами (1.3) функции φ_{1N}, φ_{2N} являются соответственно точной минорантой и точной мажорантой для функций из $F(z^N)$. Поэтому образом множества $F(z^N)$

при отображении $S(f)=\int\limits_K f(x)\,dx$ является отрезок

$$S(F(z^N))=\left[\int\limits_K \varphi_{1N}(x)\,dx,\ \int\limits_K \varphi_{2N}(x)\,dx\right].$$

По определению (см. п. 5 § 4 гл. 1) точка $\widetilde{\beta}_*(z^N)$ является центром этого отрезка, а величина $\varepsilon_N(z^N)$ — его радиусом, т. е. половиной длины. △

Замечание 1. Квадратурная формула с оптимальными коэффициентами $p_i^0(x^N)$, $i=1,\ldots,N$, не совпадает, вообще говоря, с центральной итоговой операцией $\widetilde{\beta}_*(z^N)$. Действительно, пусть $\rho(u,v)=\max\limits_{i=1,\ldots,N}|u^i-v^i|$, $K=\{u=(u^1,\ldots,u^n)\,|\,0\leqslant u^i\leqslant 1,\ i=1,\ldots,n\}$, $n=2$, $N=4$, $x_1=(1/4,\ 1/4)$, $x_2=(3/4,\ 1/4)$, $x_3=(3/4,\ 3/4)$, $x_4=(1/4,\ 3/4)$, $x^4=(x_1,x_2,x_3,x_4)$. Тогда $p_1^0(x^4)=\ldots=p_4^0(x^4)=1/4$. Нетрудно убедиться, что при $0\leqslant y\leqslant 1/2$ ситуация $z^4=(x^4,\ y,\ 0,0,0)$ реализуема (см. п. 2 § 3 гл. 1) и функция, задающая центральную итоговую операцию, принимает вид $\widetilde{\beta}_*(z^4)=-5y^3/24+3y^2/16+y/4$, тогда как $\sum\limits_{i=1}^4 p_i^0(x^4)\,y_i=y/4$, где $(y_1,y_2,y_3,y_4)=(y,\ 0,0,0)$.

В то же время, как будет показано в п. 1 § 3, при $n=1$ и произвольных $x_1,\ldots,x_N$ квадратурная формула с оптимальными коэффициентами совпадает с центральной итоговой операцией. △

Замечание 2. С помощью теоремы 1.4 легко доказывается теорема 1.3. Действительно, для произвольных фиксированных $x_1,\ldots,x_N\in K$

$$\min_{\widetilde{\beta}\in\widetilde{B}_N}\sup_{f\in F}\gamma(S(f),\ \widetilde{\beta}(x_1,\ldots,x_N,\ f(x_1),\ldots,f(x_N))=$$

$$=\sup_{y^N\in l(x^N)}\varepsilon_N(z^N)\geqslant\varepsilon_N(x^N,\ 0)=\int\limits_K\min_{i=1,\ldots,N}\rho(x,x_i)\,dx,$$

где первое равенство обосновано при доказательстве теоремы 4.3 гл. 1, а неравенство справедливо, поскольку $0\in l(x^N)$. Таким образом, никакая итоговая операция $\widetilde{\beta}\in\widetilde{B}_N$ не может гарантировать результат лучше, чем $\int\limits_K\min\limits_{i=1,\ldots,N}\rho(x,x_i)\,dx$. С другой стороны, как показывает теорема 1.2, такой результат гарантирует квадратурная

формула с коэффициентами $p_i^0(x^N)$, $i=1, \ldots, N$. Следовательно, эта квадратурная формула задает оптимальную итоговую операцию при фиксированных $x_1, \ldots, x_N$. △

Замечание 3. Теорема 1.4 справедлива, как видно из ее доказательства, не только для $F=F_\rho$, но и для произвольного функционального класса F, если только φ_{1N}, φ_{2N} — соответственно точная миноранта и точная мажоранта для функций из подкласса $F(z^N)$ класса F. △

§ 2. Построение и исследование оптимальных квадратур для функциональных классов, заданных модулями непрерывности

В этом параграфе полученные в § 1 результаты конкретизируются для функциональных классов, традиционно рассматриваемых в теории оптимальных квадратурных формул. В одних случаях удается в явном виде построить оптимальные или асимптотически оптимальные формулы, в других — лишь получить оценки на коэффициенты в выражениях для погрешности оптимальной квадратуры.

Основой для последующих построений является полученное в § 1 (см. следствие к теореме 1.2) выражение для погрешности m_N оптимальной на классе F_ρ квадратуры:

$$m_N = \inf_{x^N \in K^N} \int_K \min_{i=1,\ldots,N} \rho(x, x_i)\, dx \tag{2.1}$$

(напомним, что фигурирующие в гл. 2 функции $\rho(u, v)$ предполагаются суммируемыми по u при любом фиксированном v).

1. Вспомогательные утверждения. Введем необходимые для дальнейшего обозначения и получим некоторые вспомогательные результаты.

Для произвольной квазиметрики ρ будем называть множество $\{u \mid \rho(u, v) \leqslant R\}$ *ρ-шаром радиуса R с центром v.*

Ясно, что ρ-шары радиуса $\sup\limits_{x \in K} \min\limits_{i=1,\ldots,N} \rho(x, x_i)$ с центрами в точках $x_1, \ldots, x_N$ покрывают K и никакие ρ-шары меньшего радиуса с центрами в $x_1, \ldots, x_N$ не покрывают K. Назовем величину

$$R(x^N) \stackrel{\text{def}}{=} \sup_{x \in K} \min_{i=1,\ldots,N} \rho(x, x_i) \tag{2.2}$$

радиусом покрытия, заданного центрами $x_1, \ldots, x_N$ (*вектором* x^N). Покрытие минимального радиуса назовем *оптимальным*.

Таким образом, для отыскания оптимального покрытия надо найти величину

$$R_N \overset{\text{def}}{=} \inf_{x_1,\ldots,x_N \in K} \sup_{x \in K} \min_{i=1,\ldots,N} \rho(x, x_i), \tag{2.3}$$

называемую *радиусом оптимального покрытия*, и точки $x_1, \ldots, x_N$, при которых достигается значение, равное нижней грани.

Пусть дальше в этом параграфе $\rho(u, v) = \|u - v\|$, где $\|\cdot\|$ — произвольная норма в n-мерном координатном пространстве. Пусть $V(R)$ — объем шара радиуса R, т. е. множества $\Omega(R) = \{x \mid \|x\| \leqslant R\}$, а r_N определяется из уравнения

$$NV(r_N) = \mu(K), \tag{2.4}$$

где $\mu(K)$ по-прежнему обозначает лебегову меру множества K.

Лемма 2.1. *Справедлива оценка*

$$N \int\limits_{\Omega(r_N)} \|x\| \, dx \leqslant m_N \leqslant N \int\limits_{\Omega(R_N)} \|x\| \, dx.$$

Доказательство. Сравним лебеговы интегральные суммы для интегралов $N \int\limits_{\Omega(r_N)} \|x\| \, dx$ и

$$\int\limits_K \min_{i=1,\ldots,N} \|x - x_i\| \, dx = \sum_{i=1}^{N} \int\limits_{K_i} \|x - x_i\| \, dx = \sum_{i=1}^{N} \int\limits_{K_i'} \|x\| \, dx,$$

где x_i произвольны, $K_i = K_i(x^N)$, т. е. множества K_i образуют ρ-разбиение Вороного — Дирихле множества K (см. п. 4 § 1), $K_i' = K_i - x_i$, $i = 1, \ldots, N$.

Пусть целое число k произвольно, а j_0 таково, что

$$\frac{j_0 - 1}{k} r_N < \sup_{x \in K} \min_{i=1,\ldots,N} \|x - x_i\| = \max_{i=1,\ldots,N} \max_{x \in K_i'} \|x\| \leqslant \frac{j_0}{k} r_N.$$

Пусть, далее,

$$A_j = \{x \mid (j-1) r_N / k \leqslant \|x\| \leqslant j r_N / k\}.$$

Тогда

$$N\sum_{j=1}^{k}\frac{j}{k}r_N\mu(A_j)-\sum_{i=1}^{N}\sum_{j=1}^{j_0}\frac{j}{k}r_N\mu(A_j\cap K_i')=$$

$$=\sum_{i=1}^{N}\left[\sum_{j=1}^{k}\frac{j}{k}r_N[\mu(A_j)-\mu(A_j\cap K_i')]-\sum_{j=k+1}^{j_0}\frac{j}{k}r_N\mu(A_j\cap K_i')\right]\leqslant$$

$$\leqslant r_N\sum_{i=1}^{N}\left[\sum_{j=1}^{k}[\mu(A_j)-\mu(A_j\cap K_i')]-\sum_{j=k+1}^{j_0}\mu(A_j\cap K_i')\right]=$$

$$=r_N\sum_{i=1}^{N}[V(r_N)-\mu(K_i')]=r_N\left[NV(r_N)-\sum_{i=1}^{N}\mu(K_i')\right]=0,$$

поскольку $NV(r_N)=\sum_{i=1}^{N}\mu(K_i')=\mu(K)$. Переходя в полученном неравенстве к пределу при $k\to\infty$, имеем

$$N\int_{\Omega(r_N)}\|x\|\,dx-\int_K\min_{i=1,\ldots,N}\|x-x_i\|\,dx\leqslant 0,$$

откуда в силу (2.1) и произвольности $x_1, \ldots, x_N$ вытекает левое неравенство в утверждении леммы.

Для доказательства правого зафиксируем произвольное $\varepsilon\geqslant 0$ и такие $x_1, \ldots, x_N\in K$, что $\sup\limits_{x\in K}\min\limits_{i=1,\ldots,N}\|x-x_i\|\leqslant R_N+\varepsilon$. Тогда

$$m_N\leqslant\int_K\min_{i=1,\ldots,N}\|x-x_i\|\,dx\leqslant$$

$$\leqslant\sum_{i=1}^{N}\int_{\{x|\,\|x-x_i\|\leqslant R_N+\varepsilon\}}\|x-x_i\|\,dx=N\int_{\Omega(R_N+\varepsilon)}\|x\|\,dx.$$

В силу произвольности $\varepsilon\geqslant 0$ отсюда следует правое неравенство в утверждении леммы. $\triangle$

Лемма 2.2. *Справедлива оценка*

$$\int_{\Omega(1)}\|x\|\,dx\,\frac{\mu(K)}{V(1)}r_N\leqslant m_N\leqslant\mu(K)R_N.$$

Доказательство. Правое неравенство непосредственно следует из определения величин m_N, R_N. Для доказательства левого достаточно заметить, что

$$\int_{\Omega(r)}\|x\|\,dx=r^{n+1}\int_{\Omega(1)}\|x\|\,dx,\quad V(r)=r^nV(1),$$

и воспользоваться левым неравенством в лемме 2.1 и определением величины r_N. $\triangle$

Лемма 2.3. *Пусть $A \subset \Omega(R)$, где A — выпуклое тело с кусочно-гладкой границей, содержащее начало координат. Пусть $\|x\|$ — одна из трех норм*

$$\|x\|_0 = \max |x^i|,$$

$$\|x\|_1 = \sum_{i=1}^{n} |x^i|, \quad \|x\|_2 = \sqrt{\sum_{i=1}^{n} (x^i)^2},$$

где $x = (x^1, \ldots, x^n)$. Тогда

$$\int_A \|x\| dx \leqslant \frac{n}{n+1} R\mu(A), \qquad \int_{\Omega(R)} \|x\| dx = \frac{n}{n+1} RV(R).$$

Доказательство. Пусть $\|x\| = \|x\|_2$. Производя сферическое преобразование координат

$$x^1 = r\cos\alpha_1, \quad x^i = r\sin\alpha_1 \ldots \sin\alpha_{i-1}\cos\alpha_i,$$
$$i = 2, \ldots, n-1, \quad x^n = r\sin\alpha_1 \ldots \sin\alpha_{n-1}$$

с якобианом $r^{n-1}\sin^{n-2}\alpha_1 \sin^{n-3}\alpha_2 \ldots \sin\alpha_{n-2}$ и учитывая, что $\|x\|_2 = r$, получаем

$$\int_A \|x\| dx =$$
$$= \int_0^{2\pi} d\alpha_1 \ldots \int_0^{2\pi} d\alpha_{n-1} \int_0^{r(\alpha_1,\ldots,\alpha_{n-1})} r^n \sin^{n-2}\alpha_1 \ldots \sin\alpha_{n-2} dr =$$
$$= \frac{1}{n+1} \int_0^{2\pi} d\alpha_1 \ldots \int_0^{2\pi} d\alpha_{n-2} \int_0^{2\pi} [r(\alpha_1, \ldots \alpha_{n-1})]^{n+1} \sin^{n-2}\alpha_1 \ldots$$
$$\ldots \sin\alpha_{n-2} d\alpha_{n-1}, \quad (2.5)$$

где $r = r(\alpha_1, \ldots, \alpha_{n-1})$ — уравнение границы тела A, а пределы интегрирования выбраны для случая, когда начало координат — внутренняя точка множества A. Учитывая, что

$$\mu(A) =$$
$$= \frac{1}{n} \int_0^{2\pi} d\alpha_1 \ldots \int_0^{2\pi} d\alpha_{n-2} \int_0^{2\pi} [r(\alpha_1, \ldots, \alpha_{n-1})]^n \sin^{n-2}\alpha_1 \ldots$$
$$\ldots \sin\alpha_{n-2} d\alpha_{n-1}, \quad (2.6)$$

и производя в (2.5) оценку $[r(\alpha_1, \ldots, \alpha_{n-1})]^{n+1} \leqslant \leqslant R[r(\alpha_1, \ldots, \alpha_{n-1})]^n$, получаем требуемое неравенство. В случае $A = \Omega(R)$, очевидно, $r(\alpha_1, \ldots, \alpha_{n-1}) \equiv R$ и доказываемое равенство также немедленно следует из (2.5) и (2.6).

Если $\|x\| = \|x\|_1$, то интеграл следует заменить суммой 2^n интегралов по пересечениям A со всеми 2^n ортантами пространства $\mathbf{R}^n$. И далее, в ортанте $x^1 \geqslant 0, \ldots, x^n \geqslant 0$, например, следует произвести преобразование координат по формулам

$$x^1 = r(1-\beta_1), \quad x^i = r\beta_1 \ldots \beta_{i-1}(1-\beta_i), \quad i = 2, \ldots, n-1,$$
$$x^n = r\beta_1 \ldots \beta_{n-1}.$$

Якобиан этого преобразования равен $r^{n-1}\beta_1^{n-2} \ldots \beta_{n-2}$, $\|x\|_1 = r$.

Если же $\|x\| = \|x\|_0$, то интеграл следует заменить суммой $2n$ интегралов по пересечениям A с областями $\{x \mid \|x\|_0 = x^i\}$, $\{x \mid \|x\|_0 = -x^i\}$, $i = 1, \ldots, n$. И далее, в области $\{x \mid \|x\|_0 = x^1\}$, например, следует произвести преобразование координат по формулам

$$x^1 = r, \quad x^i = r\gamma_{i-1}, \quad i = 2, \ldots, n.$$

Якобиан этого преобразования равен r^{n-1}, $\|x\|_0 = r$.

Ясно, что и в случае $\|x\| = \|x\|_1$, и в случае $\|x\| = \|x\|_0$ доказательство утверждения леммы может быть теперь проведено аналогично доказательству в случае $\|x\| = \|x\|_2$. $\triangle$

Лемма 2.4. *Пусть $\|x\|$ — одна из трех норм $\|x\|_i$, $i = 0, 1, 2$, K — выпуклое тело с кусочно-гладкой границей. Тогда*

$$\frac{n}{n+1} r_N \mu(K) \leqslant m_N \leqslant \frac{n}{n+1} R_N \mu(K).$$

Доказательство. Левое неравенство непосредственно вытекает из лемм 2.1, 2.3. Для доказательства правого заметим, что

$$m_N \leqslant \int_K \min_{i=1,\ldots,N} \|x - x_i^0\| \, dx$$

при любых $x_1^0, \ldots, x_N^0$, в том числе в случае, когда эти точки являются центрами оптимального покрытия множества K. Для таких x_i^0 положим $K_i^0 = \{x \mid \|x - x_i^0\| =$

$= \min\limits_{j=1,\ldots,N} \|x - x_j^0\|\}$ и заметим, что $K_i^0 - x_i^0 \subset \Omega(R_N)$, $i = 1, \ldots, N$.

Применяя лемму 2.3, получим

$$\int\limits_K \min_{i=1,\ldots,N} \|x - x_i^0\| dx = \sum_{i=1}^N \int\limits_{K_i^0} \|x - x_i^0\| =$$

$$= \sum_{i=1}^N \int\limits_{K_i^0 - x_i^0} \|x\| dx \leqslant \frac{n}{n+1} R_N \mu(K). \ \triangle$$

2. Основные результаты. Перейдем к изложению основных результатов данного параграфа (теоремы 2.1—2.3). В формулировках теорем $F = F_\rho$ при $\rho(u, v) = \|u - v\|$.

Теорема 2.1 [252]. *Пусть* $K = \bigcup\limits_{i=1}^N [\Omega(R_N) + x_i^0]$, *причем*

$$\mu[(\Omega(R_N) + x_i^0) \cap (\Omega(R_N) + x_j^0)] = 0 \ \textit{при} \ i \neq j.$$

Тогда квадратурная формула с узлами x_i^0 *и коэффициентами* $p_i^0 = V(R_N) = \mu(K)/N$ *оптимальна на классе* F, *при этом* $m_N = \int\limits_{\Omega(R_N)} \|x\| dx$.

Доказательство. Утверждение об узлах оптимальной квадратуры и величине гарантированного ею результата немедленно следует из леммы 2.1, если учесть, что в предположениях теоремы $r_N = R_N$. Утверждение о коэффициентах следует из теоремы 1.2. $\triangle$

Непосредственно из теоремы 2.1 получаем следующие результаты.

Следствие 1. *Пусть*

$$K = \{x \mid 0 \leqslant x^i \leqslant 1, \ i = 1, \ldots, n\}, \quad \|x\| = \|x\|_0, \quad N = m^n.$$

Тогда точки

$$\left(\frac{j_1}{2m}, \ldots, \frac{j_n}{2m}\right), \quad j_1, \ldots, j_n \in \{1, 3, \ldots, 2m - 1\},$$

являются узлами оптимальной на классе F *квадратурной формулы, а числа* $p_i^0 = 1/m^n$ — *ее коэффициентами*, $i = 1, \ldots, m^n$. $\triangle$

Следствие 2. *Пусть*

$$n = 2, \quad K = \{x \mid |x^1| + |x^2| \leqslant 1\}, \quad \|x\| = \|x\|_1, \quad N = m^2.$$

Тогда точки

$$\left(\frac{j_1 - j_2}{2m}, \frac{j_1 + j_2 - 2m}{2m}\right), \quad j_1, j_2 \in \{1, 3, \ldots, 2m-1\},$$

являются узлами оптимальной на классе F квадратурной формулы, а числа $p_i^0 = 2/N^2$ — ее коэффициентами, $i = 1, \ldots, m^2$. $\triangle$

Решетки оптимальных узлов, соответствующие условиям следствий 1 и 2, изображены на рис. 2.

В соответствии с общими определениями п. 12 § 4 гл. 1 назовем последовательность квадратурных формул

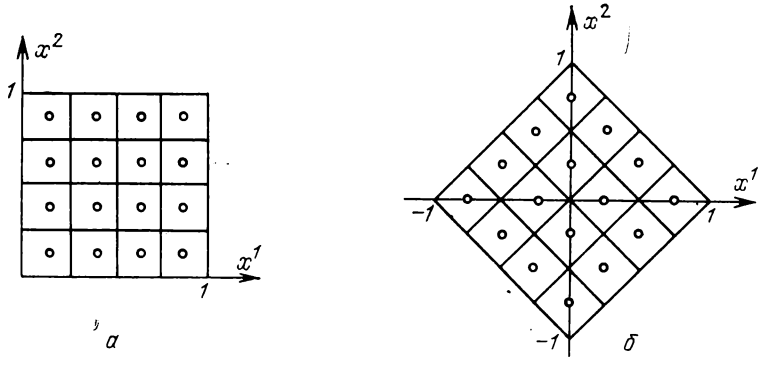

Рис. 2. Узлы оптимальной на классе F_ρ квадратурной формулы при $n = 2$, $m = 4$ в случаях: а) $\rho(u, v) = \|u - v\|_0$, б) $\rho(u, v) = \|u - v\|_1$

с узлами $x_1(N), \ldots, x_N(N)$ и коэффициентами $p_1(N), \ldots, p_N(N)$ *асимптотически оптимальной* на F при $N \to \infty$, если

$$\sup_{f \in F} \left| \int_K f(x)\,dx - \sum_{i=1}^{N} p_i(N) f(x_i(N)) \right| \sim m_N, \quad N \to \infty.$$

Теорема 2.2 [252]. *Пусть K — ограниченное множество с границей нулевой меры, норма $\|\cdot\|$ такова, что шарами, получающимися из $\Omega(1) = \{x \mid \|x\| \leqslant 1\}$ параллельными переносами, можно заполнить все пространство, т. е. для некоторых $x_1, x_2, \ldots$*

$$\bigcup_{i=1}^{\infty} (\Omega(1) + x_i) = \mathbf{R}^n, \quad \mu[(\Omega(1) + x_i) \cap (\Omega(1) + x_j)] = 0,$$

$$i \neq j.$$

Пусть $R(N)$ — минимальное из чисел R таких, что число шаров $\Omega(R)+Rx_i$, имеющих с K пересечение положительной меры, не превосходит N, а $R(N)x_{i_j}$ — центры этих шаров ($j=1, \ldots, N_1$; $N_1 \leqslant N$).

Тогда последовательность квадратурных формул с узлами $x_1(N), \ldots, x_N(N)$, где $x_j(N)=R(N)x_{i_j}$ при $j=1, \ldots, N_1$ и $x_j(N)$ — произвольные точки из K при $j=N_1+1, \ldots, N$, и коэффициентами

$$p_j(N)=\mu\{x \in K \mid \|x-x_j(N)\|=\min_{\nu=1,\ldots,N}\|x-x_\nu(N)\|\},$$
$$j=1, \ldots, N,$$

асимптотически оптимальна на F, при этом

$$m_N \sim \int\limits_{\Omega(R(N))} \|x\|\,dx \text{ при } N \to \infty.$$

Доказательство. Так как граница K нульмерна, то $NV(R(N)) \to \mu(K)=NV(r_N)$ при $N \to \infty$, и, стало быть, $R(N) \sim r_N$. Но $r_N \leqslant R_N \leqslant R(N)$ (см. (2.3)), следовательно, $r_N \sim R_N$, и по лемме 2.1 $m_N \sim N \int\limits_{\Omega(R(N))} \|x\|\,dx$. В силу теоремы 1.2 результат, гарантированный формулой с узлами $x_j(N)$ и коэффициентами $p_j(N)$, $j=1, \ldots, N$, равен $\int\limits_K \min\limits_{i=1,\ldots,N} \|x-x_i(N)\|\,dx$. Очевидно, что

$$m_N \leqslant \int\limits_K \min_{i=1,\ldots,N} \|x-x_i(N)\|\,dx \leqslant N \int\limits_{\Omega(R(N))} \|x\|\,dx.$$

С учетом уже доказанной эквивалентности левой и правой частей неравенства это завершает доказательство теоремы. △

Доказанная теорема позволяет легко строить асимптотически оптимальные квадратуры для классов, заданных такими нормами, что все пространство может быть заполнено транслятами выпуклого симметричного относительно начала координат тела $\Omega(1)=\{x \mid \|x\| \leqslant 1\}$ (его называют индикатрисой длин для метрики $\|u-v\|$). Для $\|\cdot\|_0$ и $\|\cdot\|_1$ при $n=2$ такое построение было выполнено в [8, 164].

Поскольку всякое симметричное относительно начала координат выпуклое тело задает, как известно, норму, встает вопрос, какие выпуклые тела обладают указанным свойством (с помощью параллельных переносов

заполняют пространство без пересечений ненулевой меры). Этому вопросу посвящен ряд работ. Например, при $n=3$ такие тела и, следовательно, соответствующие нормы полностью перечислены в [331] для случая, когда центры тел, заполняющих пространство, образуют решетку. О выпуклых многогранниках, обладающих этим свойством (параллелоэдрах) при $n=5$, см. [223].

Леммы 2.2 и 2.4 позволяют получить некоторые результаты об оптимальных квадратурах и в случаях, когда класс F функций, определенных на некотором множестве n-мерного координатного пространства, задан с помощью норм $\|\cdot\|_1$ при $n>2$ и $\|\cdot\|_2$. Нетрудно видеть, что в обоих случаях пространство не может быть заполнено транслятами соответствующих выпуклых тел $\Omega(1)$. Сформулируем эти результаты в виде следующей теоремы.

Теорема 2.3 [252]. *Пусть K — ограниченное множество ненулевой меры с границей меры нуль. Тогда при достаточно больших N*

$$a(n)N^{-1/n} \leqslant m_N \leqslant b(n)N^{-1/n},$$

где

$$a(n) = (\mu(K))^{(n+1)/n} \int\limits_{\Omega(1)} \|x\|\,dx/(V(1))^{(n+1)/n},$$

$$b(n) = (\mu(K))^{(n+1)/n} \sqrt[n]{n\ln n + n\ln\ln n + 5n}/(V(1))^{1/n},$$

при этом если $K=\{x \mid 0 \leqslant x^i \leqslant 1,\ i=1,\ \ldots,\ n\}$, $\|x\| = \|x\|_1$, то

$$a(n) \sim b(n) \sim \frac{n}{2e} \quad \text{при} \quad n \to \infty;$$

если же $K=\{x \mid 0 \leqslant x^i \leqslant 1,\ i=1,\ \ldots,\ n\}$, $\|x\| = \|x\|_2$, то

$$a(n) \sim b(n) \sim \frac{\sqrt{n}}{\sqrt{2\pi e}} \quad \text{при} \quad n \to \infty.$$

Доказательство. Пусть $\|x\|$ — произвольная норма. Согласно [419], при $n \geqslant 3$ справедливо следующее неравенство для наименьшей плотности θ_n (определение см., например, в [220]) покрытия n-мерного координатного пространства транслятами выпуклого тела $\Omega(1)$:

$$\theta_n \leqslant n\ln n + n\ln\ln n + 5n. \tag{2.7}$$

В силу [126, теорема 9]

$$\theta_n = \lim_{N\to\infty} NV(R_N)/\mu(K).$$

Учитывая, что $V(r)=r^nV(1)$, получаем, что при достаточно больших N

$$R_N^n < (n\ln n + n\ln\ln n + 5n)\mu(K)/(NV(1)).$$

Из определения величины r_N следует, что величина $r_N^n=\mu(K)/(NV(1))$. Из леммы 2.2 получаем теперь, что при достаточно больших N

$$\frac{(\mu(K))^{(n+1)/n}\int\limits_{\Omega(1)}\|x\|\,dx}{(V(1))^{(n+1)/n}}N^{-1/n}\leqslant m_N\leqslant$$

$$\leqslant\frac{(\mu(K))^{(n+1)/n}\sqrt[n]{n\ln n+n\ln\ln n+5n}}{(V(1))^{1/n}}N^{-1/n}. \quad (2.8)$$

Воспользовавшись выражением из леммы 2.3 для $\int\limits_{\Omega(1)}\|x\|\,dx$ и известными формулами

$$V(1)=\frac{2^n}{n!}\text{ при }\|x\|=\|x_1\|,\quad V(1)=\frac{2\pi^{n/2}}{n\Gamma(n/2)}\text{ при }\|x\|=\|x\|_2,$$

$$\sqrt[n]{\Gamma(n/2)}\sim\sqrt{n}/\sqrt{2e},\qquad \sqrt[n]{n!}\sim n/e\text{ при }n\to\infty,$$

убеждаемся в справедливости остальных утверждений теоремы. △

Замечание 1. Пусть $\|x\|$ — одна из трех норм $\|x\|_i$, $i=0,1,2$, K — выпуклое тело с кусочно гладкой границей. Тогда неравенство, аналогичное (2.8), имеет вид

$$\frac{n}{n+1}(\mu(K))^{(n+1)/n}(V(1))^{-1/n}N^{-1/n}\leqslant m_N\leqslant$$

$$\leqslant\frac{n}{n+1}(\mu(K))^{(n+1)/n}\left(V(1)\right)^{-1/n}\sqrt[n]{\theta_n^*}N^{-1/n}.$$

Неравенство справедливо при достаточно больших N. Его левая часть совпадает с левой частью (2.8), а правая часть получена так же, как в (2.8), только вместо леммы 2.2 использована лемма 2.4, а вместо оценки (2.7) некоторая оценка $\theta_n<\theta_n^*$. △

Замечание 2. Если $\|x\|=\|x\|_0$, то $\theta_n=1$. Используя предыдущее замечание, получаем в соответствующих предположениях

$$m_N\sim\frac{n}{2(n+1)}(\mu(K))^{(n+1)/n}N^{-1/n}\quad\text{при}\quad N\to\infty.\ \triangle$$

Замечание 3. Основным содержанием теоремы 2.3 является не тот факт, что $m_N \asymp N^{-1/n}$ при $N \to \infty$ (порядок погрешности при более общих предположениях найден в [13]), а формулы для $a(n)$ и $b(n)$, позволяющие получить асимптотику по размерности пространства коэффициента при $N^{-1/n}$ в выражении для погрешности m_N оптимальной квадратуры с N узлами. △

Замечание 4. Оценка погрешности $c(n)N^{-1/n}$ с $a(n) \leqslant c(n) \leqslant b(n)$, как видно из доказательства теоремы, реализуется на формулах, узлы которых задают оптимальное покрытие (связь решеток узлов оптимальных формул численного интегрирования с оптимальными упаковками была выявлена в [229]). Однако узлы оптимальной квадратурной формулы не совпадают, вообще говоря, ни с центрами оптимального покрытия, ни с центрами оптимальной упаковки. В этом нетрудно убедиться на примере

$$\|x\| = \|x\|_0,\ K = \{x = (x^1, x^2) \mid 0 \leqslant x^1,\ x^2 \leqslant 1\},$$
$$m^2 < N < (m+1)^2,$$

где m — целое. △

Замечание 5. Доказательство (2.7) в [419], так же как и вывод других известных оценок $\theta_n < \theta_n^*$, обладающих нужным нам свойством $\sqrt[n]{\theta_n^*} \sim 1$ при $n \to \infty$, неконструктивны, т. е. не дают способов построения покрытий, на которых получаемые оценки реализуются. △

Замечание 6. При построении квадратурных формул для класса F с $\|x\| = \|x\|_2$ при малых n может оказаться целесообразным, выбирая узлы интегрирования, использовать так называемую главную решетку первого типа Вороного Γ_1^n. Эта решетка задает наименее плотное решетчатое покрытие пространства при $n \leqslant 5$ [223] и обладает другими экстремальными свойствами [56, 222]. △

§ 3. Последовательно-оптимальные и оптимальные на один шаг алгоритмы численного интегрирования

Наряду с изучением квадратурных формул и пассивных алгоритмов численного интегрирования, немалое внимание уделяется и построению последовательных (адаптивных) алгоритмов (их называют еще квадратурными процессами, алгоритмами с автоматическим выбором шага). Из работ в этом направлении упомянем [15—

18, 22, 50, 67, 69, 74, 135—137, 332, 333, 339, 341, 356, 368, 379, 380, 382, 413—417, 421, 441, 442, 444]. В этом параграфе изучается одномерный случай. В § 5, 6 полученные результаты применяются для вычисления интегралов от функций, зависящих от нескольких переменных. Всюду в данном параграфе и до конца главы предполагается, если явно не оговорено противное, что $S(f)=\int\limits_K f(x)\,dx$ и $B=\mathbf{R}$ (см. п. 1 § 1 гл. 1), т. е. расстояние между элементами $S(f_1)$, $S(f_2)$ есть $|S(f_1)-S(f_2)|$.

1. Оптимальные по точности алгоритмы для классов функций, удовлетворяющих условию Липшица. Рассмотрим функциональный класс

$$F=\{f|\ |f(u)-f(v)|\leqslant M|u-v|,\ u,\ v\in[a,\ b]\},\qquad (3.1)$$

предполагая, что информационными вычислениями являются вычисления значений интегрируемой функции. Этот класс есть частный случай класса F_ρ при $K=[a,\ b]$, $\rho(u,\ v)=\|u-v\|=M|u-v|$. Из теоремы 2.1 немедленно получаем известный результат [22, 27], состоящий в том, что составная квадратурная формула прямоугольников с равноотстоящими узлами, т. е. формула, узлы и коэффициенты которой составляют векторы

$$x_0^N=\left(a+\frac{b-a}{2N},\ a+3\frac{b-a}{2N},\ \ldots,\ a+(2N-1)\frac{b-a}{2N}\right),$$
$$p_0^N=\left(\frac{b-a}{N},\ \ldots,\ \frac{b-a}{N}\right),\qquad (3.2)$$

оптимальна на классе (3.1), причем

$$m_N=N\int\limits_{-(b-a)/(2N)}^{(b-a)/(2N)} M\,|\,x\,|\,dx=\frac{M\,(b-a)^2}{4N}.\qquad (3.3)$$

С помощью теоремы 1.4 легко получить выражение для центральной итоговой операции и наилучшей гарантированной после завершения информационных вычислений точности. Для рассматриваемого класса имеем

$$\varphi_{1N}(x)=\max_{j=1,\ldots,N}\{y_j-M\,|\,x-x_j\,|\},$$
$$\varphi_{2N}(x)=\min_{j=1,\ldots,N}\{y_j+M\,|\,x-x_j\,|\},$$

$$\tilde{\beta}_*(z^N) = \int_a^b \frac{1}{2}[\varphi_{1N}(x) + \varphi_{2N}(x)]\,dx =$$

$$= (x_1 - a)\,y_1 + \sum_{j=2}^{N}(x_j - x_{j-1})\frac{y_j + y_{j-1}}{2} + (b - x_N)\,y_N, \quad (3.4)$$

$$\varepsilon_N(z^N) = \int_a^b \frac{1}{2}[\varphi_{2N}(x) - \varphi_{1N}(x)]\,dx =$$

$$= \frac{M}{2}(x_1 - a)^2 + \frac{1}{4M}\sum_{j=2}^{N}[(x_j - x_{j-1})^2 M^2 - (y_j - y_{j-1})^2] +$$

$$+ \frac{M}{2}(b - x_N)^2, \quad (3.5)$$

где предполагается, что $a \leqslant x_1 < x_2 < \ldots < x_N \leqslant b$.

Равенства (3.4), (3.5) иллюстрирует рис. 3, на котором наклонные отрезки прямых имеют угловые коэффициенты $\pm M$. Величина (3.4) равна площади фигуры,

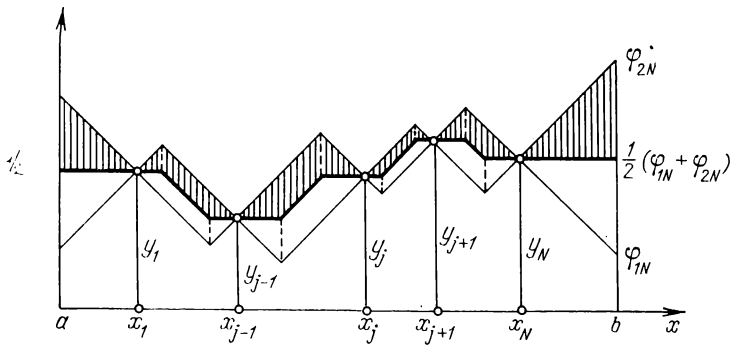

Рис. 3. Центральная итоговая операция в задачах интегрирования и восстановления функций для класса F функций, удовлетворяющих условию Липшица; миноранта φ_{1N} и мажоранта φ_{2N} для подкласса $F(z^N)$

заключенной между графиком функции $\frac{1}{2}(\varphi_{1N} + \varphi_{2N})$, который выделен жирной линией, и осью абсцисс. Величина (3.5) равна площади заштрихованной фигуры.

Вычисляя $\sup\limits_{y^N} \varepsilon_N(z^N)$ и минимизируя полученную функцию по x^N, легко непосредственно получить (3.3) и

доказать оптимальность x_0^N, определенного формулой (3.2).

Тождественно преобразуя правую часть (3.4), убеждаемся, что

$$\tilde{\beta}_*(z^N) = \left(\frac{x_1+x_2}{2} - a\right) y_1 + \sum_{j=2}^{N-1} \left(\frac{x_j+x_{j+1}}{2} - \frac{x_{j-1}+x_j}{2}\right) y_j + \\ + \left(b - \frac{x_{N-1}+x_N}{2}\right) y_N.$$

Полученное равенство показывает, что для класса (3.1) центральная итоговая операция и квадратурная формула с оптимальными коэффициентами совпадают. В рассматриваемом случае квадратурной формулой с оптимальными коэффициентами является составная квадратурная формула прямоугольников (рис. 4).

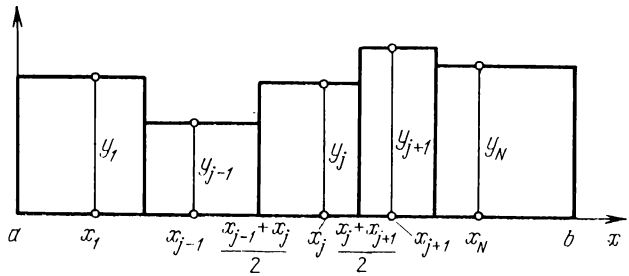

Рис. 4. Квадратурная формула с оптимальными коэффициентами — составная квадратурная формула прямоугольников

Для построения последовательно-оптимальных по точности алгоритмов окажется необходимым решить задачи отыскания оптимальных алгоритмов на подклассах класса функций (3.1) с фиксированными значениями на одном или обоих концах отрезка интегрирования $[a, b]$. Решение этих задач дают следующие две леммы, в формулировках которых $x_0 = a$, $x_{r+1} = b$, число информационных вычислений (вычислений значений интегрируемой функции) предполагается равным r, а через F обозначается функциональный класс (3.1). В этом случае $F(x_{r+1}, y_{r+1})$, $F(x_0, y_0)$, $F(x_0, x_{r+1}, y_0, y_{r+1})$ — подклассы класса F, содержащие функции с фиксированными значениями соответственно на правом, левом и обоих концах отрезка интегрирования.

Лемма 3.1. *Алгоритм* $(x_R^r, \tilde{\beta}_*(x_R^r, x_{r+1}, y^r, y_{r+1})) \in A^r$, *где*

$$x_R^r = \left(a + \frac{b-a}{2r+1},\ a + 3\frac{b-a}{2r+1},\ \ldots,\ a + (2r-1)\frac{b-a}{2r+1}\right),$$

$$\tilde{\beta}_*(x^r, x_{r+1}, y^r, y_{r+1}) = (x_1 - a)\,y_1 + \sum_{j=2}^{r+1}(x_j - x_{j-1})\frac{y_j + y_{j-1}}{2}$$

— центральная итоговая операция, оптимален по точности на классе $F(x_{r+1}, y_{r+1})$ *в множествах всех пассивных алгоритмов* A^r *и всех последовательных алгоритмов* $\tilde{A}^r$.

Алгоритм $(x_L^r, \tilde{\beta}_*(x_0, x_L^r, y_0, y^r)) \in A^r$, *где*

$$x_L^r = \left(a + 2\frac{b-a}{2r+1},\ a + 4\frac{b-a}{2r+1},\ \ldots,\ a + 2r\frac{b-a}{2r+1}\right),$$

$$\tilde{\beta}_*(x_0, x_L^r, y_0, y^r) = \sum_{j=1}^{r}(x_j - x_{j-1})\frac{y_j + y_{j-1}}{2} + (b - x_r)\,y_r$$

— центральная итоговая операция, оптимален по точности на классе $F(x_0, y_0)$ *в множествах всех пассивных алгоритмов* A^r *и всех последовательных алгоритмов* $\tilde{A}^r$.

Наилучший гарантированный результат в обоих случаях равен $\frac{M}{4}\cdot\frac{(b-a)^2}{r+1/2}$.

Доказательство. Рассмотрим функциональный класс $F(x_{r+1}, y_{r+1})$. Ясно, что подкласс всех функций из $F(x_{r+1}, y_{r+1})$, принимающих в точках $x_1, \ldots, x_r$ значения $y_1, \ldots, y_r$ соответственно, совпадает с $F(z^{r+1})$. Поскольку центральная итоговая операция обладает свойством универсальной оптимальности (см. теорему 4.1 гл. 1 и следствие из нее), достаточно доказать оптимальность x_R^r. Применяя к классу $F(x_{r+1}, y_{r+1})$ теорему 4.3 гл. 1, получаем

$$\inf_{\alpha\in A^r}\ \sup_{f\in F(x_{r+1}, y_{r+1})} \varepsilon(\alpha, f) =$$

$$= \inf_{x^r\in X^r}\ \sup_{y^r\in\{y^r \mid F(z^{r+1})\neq\varnothing\}}\ \inf_{\beta\in B}\ \sup_{f\in F(z^{r+1})} \gamma(S(f), \beta) =$$

$$= \inf_{x^r\in X^r}\ \sup_{y^r\in\{y^r \mid F(z^{r+1})\neq\varnothing\}} \varepsilon_{r+1}(z^{r+1}).$$

Далее, согласно (3.5),

$$\varepsilon_{r+1}\left(z^{r+1}\right)=\frac{M}{2}(x_1-a)^2+\frac{1}{4M}\sum_{j=2}^{r+1}[(x_j-x_{j-1})^2M^2-$$
$$-(y_j-y_{j-1})^2].$$

Нетрудно видеть, что

$$\max_{y^r\in\{y^r\mid F(z^{r+1})\neq\varnothing\}}\varepsilon_{r+1}\left(z^{r+1}\right)=\frac{M}{2}(x_1-x_0)^2+\frac{M}{4}\sum_{j=2}^{r+1}(x_j-x_{j-1})^2,$$

причем максимум достигается при $y^r=(y_{r+1},\ \ldots,\ y_{r+1})$. Таким образом, для отыскания оптимального алгоритма и наилучшего гарантированного результата остается решить задачу

$$\frac{M}{2}t_1^2+\frac{M}{4}\sum_{j=2}^{r+1}t_j^2\to\min,$$

$$t_1+\ldots+t_{r+1}=b-a,\quad t_1\geqslant 0,\ \ldots,\ t_{r+1}\geqslant 0,$$

где $t_j=x_j-x_{j-1}$, $j=1,\ \ldots,\ r+1$. Решив ее, убеждаемся в том, что алгоритм $\left(x_R^r,\ \widetilde{\beta}_*\left(x_R^r,\ x_{r+1},\ y^r,\ y_{r+1}\right)\right)$ оптимален в A^r, а наилучший гарантированный результат (т. е. значение минимума в рассматриваемой задаче на условный экстремум) равен $\frac{M}{4}\cdot\frac{(b-a)^2}{r+1/2}$. Оптимальность этого алгоритма в A^r следует из теоремы 5.2 гл. 1.

Точно так же доказываются утверждения леммы, относящиеся к классу $F(x_0,\ y_0)$. △

Лемма 3.2. *Алгоритм* $\left(x_C^r,\ \widetilde{\beta}_*\left(x_0,\ x_C^r,\ x_{r+1},\ y_0,\ y^r,\ y_{r+1}\right)\right)\in A^r$, *где*

$$x_C^r=\left(a+\frac{b-a}{r+1},\ a+2\,\frac{b-a}{r+1},\ \ldots,\ a+r\,\frac{b-a}{r+1}\right),$$

$$\widetilde{\beta}_*\left(x_0,\ x^r,\ x_{r+1},\ y_0,\ y^r,\ y_{r+1}\right)=\sum_{j=1}^{r+1}(x_j-x_{j-1})\,\frac{y_j+y_{j-1}}{2}$$

— центральная итоговая операция, оптимален по точности на классе $F(x_0,\ x_{r+1},\ y_0,\ y_{r+1})$ *в множествах всех пассивных алгоритмов* A^r *и всех последовательных алгоритмов* $\widetilde{A}^r$. *Наилучший гарантированный результат равен*

$$v_r\overset{\text{def}}{=}\frac{1}{4M}\cdot\frac{(b-a)^2M^2-(y_{r+1}-y_0)^2}{r+1}.$$

Доказательство. Так же, как и при доказательстве леммы 3.1, убеждаемся, что для отыскания результата, гарантированного алгоритмом $(x^r, \tilde{\beta}_*(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1}))$ на классе $F(x_0, x_{r+1}, y_0, y_{r+1})$, необходимо максимизировать по $y^r \in \{y^r | F(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1}) \neq \varnothing\}$ функцию

$$\varepsilon_{r+2}(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1}) = \\ = \frac{1}{4M} \sum_{j=1}^{r+1} [(x_j - x_{j-1})^2 M^2 - (y_j - y_{j-1})^2].$$

Легко видеть, что

$$\max_{y^r \in \{y^r | F(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1}) \neq \varnothing\}} \varepsilon_{r+2}(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1}) \leqslant \\ \leqslant \max_{y^r \in \mathbf{R}^r} \frac{1}{4M} \sum_{j=1}^{r+1} [(x_j - x_{j-1})^2 M^2 - (y_j - y_{j-1})^2] = \\ = \frac{M}{4} \sum_{j=1}^{r+1} (x_j - x_{j-1})^2 - \frac{1}{4M} \cdot \frac{(y_{r+1} - y_0)^2}{r+1}.$$

Подставляя в полученное неравенство $x^r = x_C^r$, выводим, что алгоритм $(x_C^r, \tilde{\beta}_*(x_0, x_C^r, x_{r+1}, y_0, y^r, y_{r+1}))$ гарантирует результат не хуже v_r.

Зафиксировав центральную итоговую операцию $\tilde{\beta}_*(x_0, x^r, x_{r+1}, y_0, y^r, y_{r+1})$, выберем произвольный алгоритм $\tilde{x}^r \in \tilde{X}^r$ (см. (3.10) гл. 1). Покажем, что с его помощью нельзя гарантировать результат лучше v_r. Положим $g(x) = y_0 + (y_{r+1} - y_0)(x - a)/(b - a)$. Пусть $x^r = (x_1, \ldots, x_r)$ есть вектор, получаемый по формулам (3.2) гл. 1 при применении алгоритма $\tilde{x}^r$ к функции g. Результат, гарантированный алгоритмом $(\tilde{x}^r, \tilde{\beta}_*)$, не лучше, чем

$$\varepsilon_{r+2}(x_0, x^r, x_{r+1}, y_0, g(x_1), \ldots, g(x_r), y_{r+1}) = \\ = \frac{1}{4M} \sum_{j=1}^{r+1} \left[(x_j - x_{j-1})^2 M^2 - \left(\frac{y_{r+1} - y_0}{b - a}\right)^2 (x_j - x_{j-1})^2\right] = \\ = \frac{1}{4M} \left[M^2 - \left(\frac{y_{r+1} - y_0}{b - a}\right)^2\right] \sum_{j=1}^{r+1} (x_j - x_{j-1})^2 \geqslant \\ \geqslant \frac{1}{4M} \left[M^2 - \left(\frac{y_{r+1} - y_0}{b - a}\right)^2\right] \frac{(b - a)^2}{r+1} = v_r.$$

Тем самым доказана оптимальность алгоритма $(x_C^r, \tilde{\beta}_*(x_0, x_C^r, x_{r+1}, y_0, y^r, y_{r+1})) \in A^r$ в множестве $\tilde{A}^r$, а следовательно, и в множестве A^r. $\triangle$

Заметим, что при доказательстве леммы 3.2 нельзя, как при доказательстве леммы 3.1, воспользоваться теоремой 5.2 гл. 1, поскольку класс $F(x_0, x_{r+1}, y_0, y_{r+1})$ при $y_0 \neq y_{r+1}$ несимметричен. Однако лемму 3.2 можно было бы доказать с помощью теоремы 5.1 гл. 1 (но не ее следствия).

2. Последовательно-оптимальный по точности алгоритм для класса функций, удовлетворяющих условию Липшица. Приступим к построению последовательно-оптимального алгоритма. Для упрощения обозначений положим $K = [0, 1]$. Согласно определению последовательно-оптимального алгоритма, на первом шаге выбор x_1 можно проводить в соответствии с любым оптимальным в $\tilde{A}^N$ алгоритмом. В нашем случае, в силу (3.2),

$$x_1 \in \left\{\frac{1}{2N}, \frac{3}{2N}, \ldots, \frac{2N-1}{2N}\right\}, \tag{3.6}$$

причем любую из точек (3.6) можно выбирать в качестве точки первого информационного вычисления.

Пусть i вычислений в точках $x_1, \ldots, x_i$ произведено, $x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания, $y_{ij} = f(x_{ij})$, $j = 1, \ldots, i$.

Так как значения, которые может принимать интегрируемая функция f в точках произвольного из отрезков $[x_{i,j-1}, x_{ij}]$ $(j = 1, \ldots, i+1,\ x_{i0} \overset{\text{def}}{=} 0,\ x_{i,i+1} \overset{\text{def}}{=} 1)$, не изменяют для рассматриваемого функционального класса множеств допустимых значений интегрируемой функции в точках остальных отрезков (эти множества ограничены минорантой φ_{1i} и мажорантой φ_{2i}), то для отыскания предписываемой последовательно-оптимальным алгоритмом точки x_{i+1} очередного вычисления достаточно оптимальным образом распределить оставшиеся $N-i$ вычислений по отрезкам $[x_{i,j-1}, x_{ij}]$, найти оптимальные пассивные (они же оптимальные последовательные) алгоритмы для функциональных классов, соответствующих этим отрезкам, и принять за x_{i+1} любую точку любого из этих алгоритмов.

Пусть оставшиеся $N-i$ информационных вычислений распределены так, что на отрезке $[x_{i,j-1}, x_{ij}]$ предписывается провести n_j вычислений, $j = 1, \ldots, i+1$, и на каждом из отрезков применяется оптимальный алгоритм.

Наилучшие гарантированные результаты для классов с фиксированным значением функции на одном из концов отрезка находятся по лемме 3.1 и равны $\frac{M}{4}\cdot\frac{x_{i1}^2}{n_1+1/2}$ для класса, соответствующего отрезку $[x_{i0}, x_{i1}]$, и $\frac{M}{4}\frac{(1-x_{ii})^2}{n_{i+1}+1/2}$ для класса, соответствующего отрезку $[x_{ii}, x_{i,i+1}]$. Наилучшие гарантированные результаты для классов с фиксированными значениями на обоих концах отрезка находятся по лемме 3.2 и равны $\frac{1}{4M}\frac{(x_{ij}-x_{i,j-1})^2M^2-(y_{ij}-y_{i,j-1})^2}{n_j+1}$, $j=2, \ldots, i$. Гарантированный результат в ситуации z^i при заданных $n_1, \ldots, n_{i+1}$ равен, очевидно, сумме наилучших гарантированных результатов для классов, соответствующих отрезкам $[x_{i,j-1}, x_{ij}]$, $j=1, \ldots, i+1$, т. е. равен

$$\frac{M}{4}\cdot\frac{x_{i1}^2}{n_1+1/2}+\frac{1}{4M}\sum_{j=2}^{i}\frac{(x_{ij}-x_{i,j-1})^2M^2-(y_{ij}-y_{i,j-1})^2}{n_j+1}+$$
$$+\frac{M}{4}\cdot\frac{(1-x_{ii})^2}{n_{i+1}+1/2}.$$

Для оптимального распределения оставшихся $N-i$ вычислений необходимо найти набор $n_1^i, \ldots, n_{i+1}^i$, реализующий минимум в задаче отыскания определенной равенством (6.7) гл. 1 величины

$$\varepsilon_N(z^i)=\min_{\substack{n_1,\ldots,n_{i+1}\in\{0,1,2,\ldots\}\\ \sum_{j=1}^{i+1}n_j=N-i}}\left[\frac{M}{4}\cdot\frac{x_{i1}^2}{n_1+1/2}+\right.$$
$$\left.+\frac{1}{4M}\sum_{j=2}^{i}\frac{(x_{ij}-x_{i,j-1})^2M^2-(y_{ij}-y_{i,j-1})^2}{n_j+1}+\frac{M}{4}\cdot\frac{(1-x_{ii})^2}{n_{i+1}+1/2}\right]. \tag{3.7}$$

После того как решение $n_1^i, \ldots, n_{i+1}^i$ найдено, за x_{i+1} можно принять любую из точек сетки, образованной объединением определенных в леммах 3.1 и 3.2 оптимальных наборов точек x_R^r при $r=n_1^i$, x_C^r при $r=n_j^i$,

$j = 2, \ldots, i$, x_L^r при $r = n_{i+1}^i$, соответствующих отрезкам $[x_{i,j-1}, x_{ij}]$, $j = 1, \ldots, i+1$, т.е.

$$x_{i+1} \in \left\{(2k_1 - 1)\frac{x_{i1}}{2n_1^i + 1} \,\middle|\, k_1 = 1, \ldots, n_1^i\right\} \cup$$
$$\cup \left(\bigcup_{j=2}^{i} \left\{x_{i,j-1} + k_j \frac{x_{ij} - x_{i,j-1}}{n_j^i + 1} \,\middle|\, k_j = 1, \ldots, n_j^i\right\}\right) \cup$$
$$\cup \left\{x_{ii} + 2k_{i+1}\frac{1 - x_{ii}}{2n_{i+1}^i + 1} \,\middle|\, k_{i+1} = 1, \ldots, n_{i+1}^i\right\},$$
$$i = 1, \ldots, N-1 \quad (3.8)$$

(если $n_j^i = 0$, то предполагается, что соответствующий набор пуст).

Леммы 3.1, 3.2 и последующие построения дают полное доказательство следующей теоремы.

Теорема 3.1 [441]. *Алгоритм* (3.6), (3.8) *последовательно-оптимален по точности на классе* (3.1) *с* $[a, b] = [0, 1]$. △

Замечание. Все последовательно-оптимальные алгоритмы, в соответствии с определением, в любой ситуации наилучшим образом используют всю имеющуюся информацию о функции f, т. е. наилучшим образом «наказывают» природу, в распоряжении которой находится «выбор» f, за совершенные «ошибки» — выбор не наихудших для вычислителя возможностей. Однако последовательно-оптимальные алгоритмы (3.6), (3.8) неравноценны в том смысле, что создают разные возможности для ошибок природы. Так, легко убедиться, что алгоритм $x_1 = 1/(2N)$, $x_2 = 3/(2N)$, ..., $x_N = (2N-1)/(2N)$ последовательно-оптимален по точности. В то же время этот алгоритм вообще не дает природе возможностей совершать ошибки, которые можно было бы использовать. Поэтому имеющимся в формулах (3.6), (3.8) произволом желательно разумно распорядиться. Можно, например, выбирать точку случайно равновероятно из набора допустимых точек (3.6), (3.8). Другой способ состоит в выборе из набора допустимых точек (3.6), (3.8) точки, ближайшей к точке, указываемой оптимальным на один шаг алгоритмом соответственно на первом и $(i+1)$-м шаге. Точки x_1, x_{i+1} при $i \geqslant 1$, указываемые оптимальным на один шаг алгоритмом, будут построены в п. 4 (см. (3.14), (3.15)). △

3. Решение вспомогательных задач целочисленной оптимизации. Применяя последовательно-оптимальный алгоритм на каждом шаге, приходится решать нелинейную задачу целочисленной оптимизации на отыскание $\varepsilon_N(z^i)$. Эти задачи имеют вид

$$\sum_{j=1}^{m} \frac{A_j}{n_j+\alpha_j} \to \min_{n_1,\ldots,n_m}, \quad n_1,\ldots,n_m \in \{0,1,2,\ldots\},$$

$$\sum_{j=1}^{m} n_j = p, \qquad (3.9)$$

где $A_j > 0$, $\alpha_j = 1/2$ или $\alpha_j = 1$. Разработанный в [354] (см. также [224, гл. 4] и [255, гл. 7]) алгоритм решения опирается на свойство выпуклости функций $\varphi_j(n_j) = A_j/(n_j+\alpha_j)$ и предназначен для решения задач вида

$$\sum_{j=1}^{m} \varphi_j(n_j) \to \min,$$

$$\sum_{j=1}^{m} n_j = p, \; n_j \in \{0,1,\ldots,p\}, \; j=1,\ldots,m, \qquad (3.10)$$

где для всех $k=1,\ldots,p-1$, $j=1,\ldots,m$,

$$\varphi_j(k) - \varphi_j(k-1) \leqslant \varphi_j(k+1) - \varphi_j(k). \qquad (3.11)$$

Очевидно, что условия (3.11) выполнены для выпуклых на отрезке $[0, p]$ функций $\varphi_1, \ldots, \varphi_m$. Эти условия являются аналогом определения выпуклости для функций целочисленного аргумента. Задача (3.9), следовательно, является частным случаем задачи (3.10). Отметим, что целочисленным задачам минимизации суммы выпуклых функций посвящены, помимо [354], работы [398, 424, 453] и др.

Для обоснования алгоритма решения задачи (3.10) нам потребуется произвести некоторые простые преобразования. Учитывая, что

$$\varphi_j(n_j) = (\varphi_j(n_j) - \varphi_j(n_j-1)) + (\varphi_j(n_j-1) - \varphi_j(n_j-2)) + \ldots + (\varphi_j(1) - \varphi_j(0)) + \varphi_j(0),$$

можем записать

$$\sum_{j=1}^{n} \varphi_j(n_j) = \sum_{j\in\{j \mid 1<j<m, n_j>0\}} \sum_{k=1}^{n_j} a_{jk} + C_s \qquad (3.12)$$

где

$$a_{jk} = \varphi_j(k) - \varphi_j(k-1), \quad j = 1, \ldots, m, \; k = 1, \ldots, p,$$

$$C = \sum_{j=1}^{m} \varphi_j(0).$$

Итак, исходная задача (3.10) сводится к минимизации двойной суммы (3.10), содержащей p элементов матрицы (a_{jk}). Принимая во внимание неравенства (3.11), которые можно переписать в виде

$$a_{j1} \leqslant a_{j2} \leqslant \ldots \leqslant a_{jp}, \quad j = 1, \ldots, m,$$

нетрудно видеть, что для решения задачи (3.10) достаточно из матрицы $\begin{pmatrix} a_{11} \ldots a_{1p} \\ \cdot \; \cdot \; \cdot \; \cdot \; \cdot \; \cdot \\ a_{m1} \ldots a_{mp} \end{pmatrix}$ выбрать p наименьших элементов. При этом, если в j-й строке выбрано n_j элементов, $j = 1, \ldots, m$, то $(n_1, \ldots, n_m)$ — решение задачи (3.10). Проведенное рассуждение, по существу, доказывает следующее утверждение.

Лемма 3.3 (критерий Гросса [354]). *Выполнение условия*

$$\max_{j \in \{j \mid 1 \leqslant j \leqslant m, n_j > 0\}} [\varphi_j(n_j) - \varphi_j(n_j - 1)] \leqslant$$

$$\leqslant \min_{j=1,\ldots,m} [\varphi_j(n_j + 1) - \varphi_j(n_j)]$$

необходимо и достаточно для того, чтобы вектор $(n_1, \ldots, n_m)$, *удовлетворяющий ограничениям задачи* (3.10), (3.11), *был ее решением.*

Доказательство. Приведенное условие, с учетом неравенств (3.11), показывает, что наибольший из выбранных в матрице $(a_{jk}) = (\varphi_j(k) - \varphi_j(k-1))$ элементов не превышает наименьший из невыбранных элементов, т. е. выбрано действительно p наименьших элементов. △

Запишем теперь, пользуясь оператором := (присвоить значение) следующий алгоритм.

Алгоритм 1.

Шаг 0: $n_j := 0$, $j = 1, \ldots, m$.

Шаг s $(1 \leqslant s \leqslant p)$: $\mu := \arg \min_{j=1,\ldots,m} (\varphi_j(n_j + 1) - \varphi_j(n_j))$,

$$n_\mu := n_\mu + 1. \; \triangle$$

Учитывая (3.11), убеждаемся, что совершив p раз шаг s, выберем тем самым p наименьших элементов в

сумме (3.12) и, следовательно, получим решение задачи (3.10).

Если известно хорошее начальное приближение $n_1^0, \ldots, n_m^0$, $\sum_{j=1}^{m} n_j^0 = p$, то может оказаться разумным использовать несколько иной алгоритм.

Алгоритм 2.

Шаг 0: $n_j := n_j^0$, $j = 1, \ldots, m$.

Шаг 1: $\mu := \arg \min_{j=1,\ldots,m} (\varphi_j(n_j + 1) - \varphi_j(n_j))$,

$$\nu := \arg \max_{j \in \{j \mid 1 \leqslant j \leqslant m, n_j > 0\}} (\varphi_j(n_j) - \varphi_j(n_j - 1));$$

если $\varphi_\mu(n_\mu + 1) - \varphi_\mu(n_\mu) \geqslant \varphi_\nu(n_\nu) - \varphi_\nu(n_\nu - 1)$, то СТОП, в противном случае выполнить операторы $n_\mu := n_\mu + 1$, $n_\nu := n_\nu - 1$ и повторить шаг 1. △

Нетрудно видеть, что алгоритм 2 также позволяет выбрать p наименьших элементов из матрицы (a_{jk}), т. е. получить решение задачи (3.10). Начальное приближение для задачи (3.9) можно получить, используя решение целочисленной задачи предыдущего шага или округляя до целочисленного решение задачи

$$\sum_{j=1}^{m} \frac{A_j}{n_j + \alpha_j} \to \min_{n_1,\ldots,n_m : \sum_{j=1}^{m} n_j = p}, \tag{3.13}$$

которое нетрудно получить в явном виде:

$$n_j^* = \frac{\sqrt{A_j}\,(p + \alpha)}{A} - \alpha_j, \quad j = 1, \ldots, m,$$

где

$$\alpha = \sum_{j=1}^{m} \alpha_j, \qquad A = \sum_{j=1}^{m} \sqrt{A_j}.$$

Равенство $A = 0$ означает, что точное значение интеграла $\int_0^1 f(x)\,dx$ уже найдено и решать задачу (3.13) не нужно.

Нетрудно заметить, что при $i = 1$, т. е. при выборе точки x_2, решение соответствующей задачи целочисленной оптимизации тривиально и

$$x_2 \in \left\{\frac{1}{2N}, \frac{3}{2N}, \ldots, \frac{2N-1}{2N}\right\} \setminus \{x_i\}.$$

4. Оптимальный на один шаг алгоритм для класса функций, удовлетворяющих условию Липшица. Полученные результаты позволяют также без труда построить алгоритм, обладающий свойством оптимальности на один шаг. Для оптимального выбора точки x_1 достаточно решить задачу минимизации гарантированного после первого шага результата (3.5):

$$\varepsilon_1(z^1) = \frac{M}{2}x_1^2 + \frac{M}{2}(1-x_1)^2 \to \min, \quad x_1 \in [0, 1],$$

откуда

$$x_1 = 1/2. \tag{3.14}$$

Для отыскания точки x_{i+1} при $i \geqslant 1$ необходимо решить задачу (3.7) для $N-i=1$. Иными словами, нам следует решить задачи (3.9) при $m=i+1$, $p=1$, $A_1 = Mx_{i1}^2/4$, $\alpha_1 = 1/2$, $A_{i+1} = M(1-x_{ii})^2/4$, $\alpha_{i+1} = 1/2$, $A_j = [(x_{ij}-x_{i,j-1})^2M^2 - (y_{ij}-y_{i,j-1})^2]/(4M)$, $\alpha_j = 1$, $j=2, \ldots, i$, или (3.10) с $\varphi_j(n_j) = A_j/(n_j+\alpha_j)$. Применяя для ее решения алгоритм 1, в соответствии с (3.8) получаем

$$x_{i+1} = \begin{cases} \dfrac{x_{i1}}{3}, & \text{если } \dfrac{A_1}{1+\alpha_1} - \dfrac{A_1}{\alpha_1} = m, \\ \dfrac{x_{i,k-1}+x_{ik}}{2}, & \text{если } \dfrac{A_k}{1+\alpha_k} - \dfrac{A_k}{\alpha_k} = m, k \in \{2, \ldots, i\}, \\ x_{ii} + \dfrac{2}{3}(1-x_{ii}), & \text{если } \dfrac{A_{i+1}}{1+\alpha_{i+1}} - \dfrac{A_{i+1}}{\alpha_{i+1}} = m, \end{cases} \tag{3.15}$$

где $m = \min\limits_{j=1,\ldots,i+1} \{A_j/(1+\alpha_j) - A_j/\alpha_j\}$. Если формула (3.15) определяет точку x_{i+1} не единственным образом, то выбирается любая из возможных точек. Тем самым оптимальный на один шаг алгоритм построен.

Теорема 3.2. *Алгоритм* (3.14), (3.15) *оптимален на один шаг на классе* (3.1) *с* $[a, b] = [0, 1]$. △

5. Последовательно-оптимальный по числу информационных вычислений алгоритм для класса функций, удовлетворяющих условию Липшица. Пусть точность ε фиксирована. Опишем вкратце процесс построения алгоритма, последовательно-оптимального по числу информационных вычислений на классе (3.1) при $[a, b] = [0, 1]$.

В силу теоремы 5.2 гл. 1 для рассматриваемого функционального класса наилучший гарантированный в множестве всех последовательных алгоритмов результат ε_N

совпадает с наилучшим гарантированным в множестве пассивных алгоритмов результатом m_N, причем, согласно (3.3), $m_N = M/(4N)$. Поэтому

$$N_\varepsilon = \min\{N \mid \varepsilon_N \leqslant \varepsilon\} = \min\left\{N \,\middle|\, \frac{M}{4N} \leqslant \varepsilon\right\} = \left\lceil \frac{M}{4\varepsilon} \right\rceil. \quad (3.16)$$

В соответствии с определениями оптимального по числу информационных вычислений алгоритма (см. п. 13 § 4 гл. 1) и последовательно-оптимального по числу информационных вычислений алгоритма (см. п. 4 § 6 гл. 1) выбор точки x_1 последовательно-оптимальным алгоритмом определяется формулой (3.6) при $N = N_\varepsilon$.

Если после i шагов ($i \geqslant 1$) в ситуации z^i критерий остановки (4.20) гл. 1 не выполнен, то с помощью формулы (3.7) определяется число $N_\varepsilon(z^i)$ (см. (6.9) гл. 1) и далее по формуле (3.8) находится точка x_{i+1}. При этом $n_1^i, \ldots, n_{i+1}^i$ есть решение задачи (3.7) при $N = N_\varepsilon(z^i) + i$. Тем самым построение алгоритма завершено.

Пусть точность ε по-прежнему фиксирована. Выделим из множества последовательно-оптимальных по числу информационных вычислений алгоритмов один алгоритм, которому соответствуют тривиально решаемые задачи вида (3.9) и который, следовательно, отличается простотой программной реализации.

Алгоритм определяется следующим условием. На каждом шаге выбирается самая левая точка из множества точек, допустимых для последовательно-оптимального по числу информационных вычислений алгоритма.

Таким образом (см. (3.16)),

$$x_1 = \frac{1}{2N_\varepsilon} = \frac{1}{2\lceil M/(4\varepsilon)\rceil}. \quad (3.17)$$

Для отыскания точки x_2 необходимо решить задачу вида (3.7)

$$\frac{M}{4}\cdot\frac{x_1^2}{n_1 + 1/2} + \frac{M}{4}\cdot\frac{(1-x_1)^2}{n_2 + 1/2} \to \min_{n_1, n_2},$$

$$n_1,\ n_2 \in \{0,\ 1,\ 2,\ \ldots\},\quad n_1 + n_2 = N_\varepsilon(z^1).$$

Покажем, что $(0,\ N_\varepsilon(z^1))$ — решение задачи. Для этого достаточно, в силу леммы 3.3, проверить справедливость неравенства

$$\frac{\frac{M}{4}x_1^2}{3/2} - \frac{\frac{M}{4}x_1^2}{1/2} \geqslant \frac{\frac{M}{4}(1-x_1)^2}{N_\varepsilon(z^1) + 1/2} - \frac{\frac{M}{4}(1-x_1)^2}{N^\varepsilon(z^1) - 1/2}.$$

Справедливость этого неравенства следует из (3.17) и (6.12) гл. 1. Итак, в соответствии с (3.8) и условием, определяющим алгоритм,

$$x_2 = x_1 + 2\frac{1 - x_1}{2N_\varepsilon(z^1) + 1}. \tag{3.18}$$

(Можно убедиться, что $N_\varepsilon(z^1) = N_\varepsilon - 1$, и поэтому $x_2 = 3/(2N_\varepsilon)$.)

Предположим, что $x_1 < x_2 < \ldots < x_{i-1}$ и решение задачи вида (3.9):

$$\frac{M}{4}\cdot\frac{x_1^2}{n_1 + 1/2} + \frac{1}{4M}\sum_{j=2}^{i-1}\frac{(x_j - x_{j-1})^2M^2 - (y_j - y_{j-1})^2}{n_j + 1} + \\ + \frac{M}{4}\cdot\frac{(1 - x_{i-1})^2}{n_i + 1/2} \to \min_{n_1,\ldots,n_i},$$

$$n_1, \ldots, n_i \in \{0, 1, 2, \ldots\}, \quad \sum_{j=1}^{i} n_j = N_\varepsilon(z^{i-1}),$$

имеет вид $(0, \ldots, 0, N_\varepsilon(z^{i-1}))$. В этом случае, согласно определению алгоритма,

$$x_i = x_{i-1} + 2\frac{1 - x_{i-1}}{2N_\varepsilon(z^{i-1}) + 1} \tag{3.19}$$

и в силу леммы 3.3

$$\frac{\frac{M}{4}x_1^2}{3/2} - \frac{\frac{M}{4}x_1^2}{1/2} \geqslant \frac{\frac{M}{4}(1 - x_{i-1})^2}{N_\varepsilon(z^{i-1}) + 1/2} - \frac{\frac{M}{4}(1 - x_{i-1})^2}{N_\varepsilon(z^{i-1}) - 1/2}, \tag{3.20}$$

$$\frac{\frac{1}{4M}\left[(x_j - x_{j-1})^2M^2 - (y_j - y_{j-1})^2\right]}{2} - \\ - \frac{\frac{1}{4M}\left[(x_j - x_{j-1})^2M^2 - (y_j - y_{j-1})^2\right]}{1} \geqslant \frac{\frac{M}{4}(1 - x_{i-1})^2}{N_\varepsilon(z^{i-1}) + 1/2} - \\ - \frac{\frac{M}{4}(1 - x_{i-1})^2}{N_\varepsilon(z^{i-1}) - 1/2}, \quad j = 2, \ldots, i - 1. \tag{3.21}$$

Докажем, что решение задачи

$$\frac{M}{4}\cdot\frac{x_1^2}{n_1+1/2}+\frac{1}{4M}\sum_{j=2}^{i}\frac{(x_j-x_{j-1})^2M^2-(y_j-y_{j-1})^2}{n_j+1}+$$

$$+\frac{M}{4}\cdot\frac{(1-x_i)^2}{n_{i+1}+1/2}\to\min_{n_1,\ldots,n_{i+1}},\qquad(3.22)$$

$$n_1,\ldots,n_{i+1}\in\{0,1,2,\ldots\},\quad\sum_{j=1}^{i+1}n_j=N_\varepsilon(z^i),$$

имеет вид $(0,\ldots,0,N_\varepsilon(z^i))$. При этом, естественно, предполагается, что $N_\varepsilon(z^i)\geqslant 1$ (противное означает, что требуемая точность ε уже достигнута). Для доказательства достаточно, в силу леммы 3.3, проверить справедливость неравенств

$$\frac{\frac{M}{4}x_1^2}{3/2}-\frac{\frac{M}{4}x_1^2}{1/2}\geqslant\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)+1/2}-\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)-1/2},\qquad(3.23)$$

$$\frac{\frac{1}{4M}\left[(x_j-x_{j-1})^2M^2-(y_j-y_{j-1})^2\right]}{2}-$$

$$-\frac{\frac{1}{4M}\left[(x_j-x_{j-1})^2M^2-(y_j-y_{j-1})^2\right]}{1}\geqslant$$

$$\geqslant\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)+1/2}-\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)-1/2},\quad j=2,\ldots,i.\qquad(3.24)$$

Перепишем (3.19) в виде

$$\frac{M}{4}(1-x_i)^2=\frac{M}{4}(1-x_{i-1})^2\left(\frac{N_\varepsilon(z^{i-1})-1/2}{N_\varepsilon(z^{i-1})+1/2}\right)^2.$$

Отсюда, учитывая, что $N_\varepsilon(z^{i-1})\geqslant N_\varepsilon(z^i)+1\geqslant 2$ (см. (6.12) гл. 1), получаем

$$\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)+1/2}-\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)-1/2}\leqslant\frac{-\frac{M}{4}(1-x_{i-1})^2\left(\frac{N_\varepsilon(z^{i-1})-1/2}{N_\varepsilon(z^{i-1})+1/2}\right)^2}{\left(N_\varepsilon(z^{i-1})-1/2\right)\left(N_\varepsilon(z^{i-1})-3/2\right)}\leqslant$$

$$\leqslant \frac{-\frac{M}{4}(1-x_{i-1})^2}{\left(N_\varepsilon(z^{i-1})-1/2\right)\left(N_\varepsilon(z^{i-1})+1/2\right)} = \frac{\frac{M}{4}(1-x_{i-1})^2}{N_\varepsilon(z^{i-1})+1/2} - \frac{\frac{M}{4}(1-x_{i-1})^2}{N_\varepsilon(z^{i-1})-1/2} \quad (3.25)$$

(второе неравенство в цепочке проверяется непосредственно). Теперь справедливость (3.24) при $j=2, \ldots, i-1$ и (3.23) следует из (3.21) и (3.20) соответственно. Остается проверить (3.24) при $j=i$.

Имеем

$$\frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)+1/2} - \frac{\frac{M}{4}(1-x_i)^2}{N_\varepsilon(z^i)-1/2} \leqslant \frac{-\frac{M}{4}(1-x_{i-1})^2\left(N_\varepsilon(z^{i-1})-1/2\right)}{\left(N_\varepsilon(z^{i-1})+1/2\right)^2\left(N_\varepsilon(z^{i-1})-3/2\right)} \leqslant$$

$$\leqslant \frac{-\frac{M}{4}(1-x_{i-1})^2}{2\left(N_\varepsilon(z^{i-1})+1/2\right)^2} = -\frac{M}{8}(x_i-x_{i-1})^2 \leqslant$$

$$\leqslant -\frac{1}{8M}\left[(x_i-x_{i-1})^2M^2-(y_i-y_{i-1})^2\right] =$$

$$= \frac{\frac{1}{4M}\left[(x_i-x_{i-1})^2M^2-(y_i-y_{i-1})^2\right]}{2} - \frac{\frac{1}{4M}\left[(x_i-x_{i-1})^2M^2-(y_i-y_{i-1})^2\right]}{1},$$

где первое неравенство совпадает с первым неравенством в (3.25), второе неравенство проверяется непосредственно, а справедливость первого равенства следует из (3.19). Тем самым справедливость неравенств (3.23), (3.24) установлена и, следовательно, доказано по индукции, что для любого i такого, что $N_\varepsilon(z^i) \geqslant 1$, решение задачи (3.22) имеет вид $(0, \ldots, 0, N_\varepsilon(z^i))$. Поэтому, в соответствии с формулой (3.8) и определением алгоритма,

$$x_{i+1} = x_i + 2\frac{1-x_i}{2N_\varepsilon(z^i)+1}. \quad (3.26)$$

При этом, согласно формуле (6.9) гл. 1, число $N_\varepsilon(z^i)$

определяется следующим образом:

$$N_\varepsilon(z^i) = \min\left\{N \left| \frac{Mx_1^2}{2} + \frac{1}{4M}\sum_{j=2}^{i}[(x_j - x_{j-1})^2 M^2 - \right.\right.$$

$$\left.\left. - (y_j - y_{j-1})^2] + \frac{M}{4}\cdot\frac{(1-x_i)^2}{N+1/2} \leqslant \varepsilon\right\}.$$

Нетрудно видеть, что

$$N_\varepsilon(z^i) = \left[\frac{\frac{M}{4}(1-x_i)^2}{\varepsilon - \frac{Mx_1^2}{2} - \frac{1}{4M}\sum_{j=2}^{i}\left[(x_j - x_{j-1})^2 M^2 - (y_j - y_{j-1})^2\right]}\right] - \frac{1}{2}. \tag{3.27}$$

Таким образом, полностью доказана следующая теорема.

Теорема 3.3. *Алгоритм* (3.17), (3.26), (3.27) *последовательно-оптимален по числу информационных вычислений на классе* (3.1) *с* $[a, b] = [0, 1]$. △

Как отмечалось, преимущество построенного алгоритма — простота программной реализации и малые затраты машинного времени на «внутренние потребности» алгоритма. В то же время при использовании алгоритма «природа» имеет меньше возможностей для «совершения ошибок» по сравнению с возможностями, имеющимися при использовании других последовательно-оптимальных алгоритмов (см. по этому поводу замечание к теореме 3.1).

6. Оптимальные и последовательно-оптимальные алгоритмы для класса монотонных функций. Развитые методы позволяют решить задачу построения последовательно-оптимальных алгоритмов и для классов совершенно иной, нежели класс (3.1), природы.

Рассмотрим класс

$$F = \{f \mid f \text{ не убывает на } [a, b],\ f(a) = y_0,\ f(b) = y_{N+1}\} \tag{3.28}$$

монотонных на отрезке $[a, b]$ функций с фиксированными значениями на концах отрезка, предполагая, что информационными вычислениями являются вычисления значений интегрируемой функции. Заметим, что класс (3.28) содержит не только негладкие функции, как класс (3.1), но также и разрывные функции. Этот класс изучался в работах [67, 372].

Пусть значения функции f из класса (3.28) вычислены в точках $x_1, x_2, \ldots, x_N$, $x_1 < x_2 < \ldots < x_N$, и оказались равными $y_1, y_2, \ldots, y_N$ соответственно. Очевидно, что функции (рис. 5)

$$\varphi_{1N}(x) = \begin{cases} y_{j-1} & \text{при} \quad x_{j-1} \leqslant x < x_j, \\ y_{N+1} & \text{при} \quad x = x_{N+1}, \end{cases}$$

$$\varphi_{2N}(x) = \begin{cases} y_0 & \text{при} \quad x = x_0, \\ y_j & \text{при} \quad x_{j-1} < x \leqslant x_j, \end{cases}$$

где $x_0 = a$, $x_{N+1} = b$, $j = 1, \ldots, N+1$, являются соответственно точной минорантой и точной мажорантой для функций $f \in F(z^N) = \{f \in F \mid f(x_j) = y_j,\ j = 1, \ldots, N\}$. Через

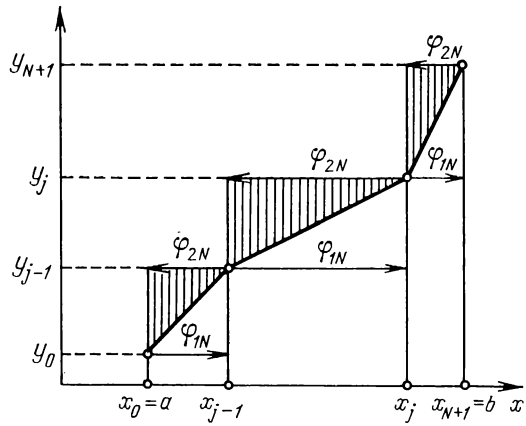

Рис. 5. Центральная итоговая операция в задаче интегрирования для класса F монотонных функций; миноранта φ_{1N} и мажоранта φ_{2N} для подкласса $F(z^N)$

миноранту и мажоранту легко выражается центральная итоговая операция (см. замечание 3 к теореме 1.4):

$$\tilde{\beta}_*(z^N) = \int_a^b \frac{1}{2}(\varphi_{1N}(x) + \varphi_{2N}(x))\,dx =$$

$$= \sum_{j=1}^{N+1} (x_j - x_{j-1}) \frac{y_j + y_{j-1}}{2}. \qquad (3.29)$$

Величина $\tilde{\beta}_*(z^N)$ равна площади фигуры, составленной из $N+1$ трапеций и заключенной между жирной линией

на рис. 5 и осью абсцисс (такая же площадь заключена, очевидно, между осью абсцисс и графиком функции $\frac{1}{2}(\varphi_{1N}+\varphi_{2N})$). Таким образом, центральная итоговая операция совпадает с составной квадратурной формулой трапеций с узлами $x_0=a,\ x_1,\ \ldots,\ x_N,\ x_{N+1}=b$. Гарантированная точность в ситуации z^N выражается по формуле (см. замечание 3 к теореме 1.4)

$$\varepsilon_N(z^N)=\int_a^b \frac{1}{2}(\varphi_{2N}(x)-\varphi_{1N}(x))\,dx=$$
$$=\sum_{j=1}^{N+1}(x_j-x_{j-1})\frac{y_j-y_{j-1}}{2}. \quad (3.30)$$

Эта величина равна площади заштрихованной на рис. 5 фигуры.

Теорема 3.4. *Алгоритм* $\alpha_0=(x_0^N,\tilde{\beta}_*)$ *при* $x_0^N=(x_1^0,\ \ldots,x_N^0)$, $x_j^0=a+j(b-a)/(N+1)$, $j=1,\ \ldots,\ N$ (*т. е. составная квадратурная формула трапеций с равноотстоящими узлами* $x_0^0=a, x_1^0,\ \ldots,x_N^0, x_{N+1}^0=b$) *оптимален по точности в множествах всех квадратурных формул с* N *узлами, всех пассивных алгоритмов* A^N *и всех последовательных алгоритмов* $\tilde{A}^N$. *Наилучший гарантированный результат равен* $\frac{(y_{N+1}-y_0)(b-a)}{2(N+1)}$.

Доказательство. Для произвольного последовательного алгоритма $\alpha=(\tilde{x}^N,\tilde{\beta})\in\tilde{A}^N$ (см. (3.11) гл. 1) и произвольной реализуемой при применении алгоритма α ситуации z^N (см. п. 2 § 3 гл. 1) имеем

$$\sup_{f\in F}\varepsilon(\alpha,f)\geqslant\sup_{f\in F(z^N)}\varepsilon(\alpha,f)=\sup_{f\in F(z^N)}|S(f)-\tilde{\beta}(z^N)|\geqslant$$
$$\geqslant\inf_{\beta\in\mathbf{R}}\sup_{f\in F(z^N)}|S(f)-\beta|=\frac{1}{2}\Big(\sup_{f\in F(z^N)}S(f)-\inf_{f\in F(z^N)}S(f)\Big)=$$
$$=\frac{1}{2}(S(\varphi_{2N})-S(\varphi_{1N})), \quad (3.31)$$

где, напомним, $S(f)=\int_a^b f(x)\,dx$. Пусть $x_1,\ \ldots,\ x_N$ — точки, полученные при применении (см. (3.2) гл. 1) алгоритма α к функции $g(x)=y_0+(y_{N+1}-y_0)(x-a)/(b-a)$. Чтобы не усложнять обозначений, будем считать, что $x_1\leqslant$

$\leqslant x_2 \leqslant \ldots \leqslant x_N$. Положим $y_j = g(x_j)$, $j = 1, \ldots, N$. Тогда $S(\varphi_{2N}) - S(\varphi_{1N}) =$

$$= \sum_{j=1}^{N+1} (y_j - y_{j-1})(x_j - x_{j-1}) = \sum_{j=1}^{N+1} (g(x_j) - g(x_{j-1}))(x_j - x_{j-1}) =$$

$$= \frac{y_{N+1} - y_0}{b-a} \sum_{j=1}^{N+1} (x_j - x_{j-1})^2 \geqslant \frac{(y_{N+1} - y_0)(b-a)}{N+1},$$

где для проверки неравенства достаточно решить методом множителей Лагранжа следующую задачу на условный экстремум:

$$\sum_{j=1}^{N+1} (x_j - x_{j-1})^2 \to \min, \quad \sum_{j=1}^{N+1} (x_j - x_{j-1}) = b - a.$$

Отсюда и из (3.31) получаем, что для произвольного $\alpha \in \tilde{A}^N$

$$\sup_{f \in F} \varepsilon(\alpha, f) \geqslant \frac{(y_{N+1} - y_0)(b-a)}{2(N+1)}. \tag{3.32}$$

Тем самым мы показали, что никакой алгоритм $\alpha \in \tilde{A}^N$ не позволяет гарантировать точность лучше $\frac{(y_{N+1} - y_0)}{2(N+1)} \times$ $\times (b-a)$.

Остается показать, что алгоритм $\alpha_0 = (x_0^N, \tilde{\beta}_*)$ гарантирует эту точность. Действительно,

$$\sup_{f \in F} \varepsilon(\alpha_0, f) = \sup_{y^N} \sup_{f \in F(x_0^N, y^N)} \varepsilon(\alpha_0, f) =$$

$$= \sup_{y^N} \sup_{f \in F(x_0^N, y^N)} |S(f) - \tilde{\beta}_*(x_0^N, y^N)| = \sup_{y^N} \varepsilon_N(x_0^N, y^N) =$$

$$= \sup_{y^N} \sum_{j=1}^{N+1} (x_j^0 - x_{j-1}^0) \frac{y_j - y_{j-1}}{2} = \frac{(y_{N+1} - y_0)(b-a)}{2(N+1)}, \tag{3.33}$$

где верхняя грань по y^N берется всюду по множеству $\{y^N \mid F(x_0^N, y^N) \neq \varnothing\}$; первое равенство справедливо, так как $F = \bigcup_{y^N} F(x_0^N, y^N)$, во втором равенстве использованы определения критерия $\varepsilon(\alpha_0, f)$ и алгоритма α_0, третье основано на соотношениях (4.3) и (4.4) гл. 1, четвертое — на формуле (3.30), наконец пятое — на определении точек $x_0^0, x_1^0, \ldots, x_N^0, x_{N+1}^0$. Из (3.32) и (3.33) сле-

дует, что алгоритм α_0 оптимален в множестве всех последовательных алгоритмов $\tilde{A}^N$, а значит, и в множествах всех квадратурных формул с N узлами и всех пассивных алгоритмов A^N. Наилучший гарантированный результат в каждом из этих множеств алгоритмов равен результату $\frac{(y_{N+1}-y_0)(b-a)}{2(N+1)}$, гарантированному оптимальным алгоритмом α_0. △

Отметим, что взамен только что изложенного можно было бы дать доказательство, опирающееся на теоремы 4.2 и 5.1 гл. 1.

Построение последовательно-оптимального алгоритма происходит по той же схеме, что и для класса (3.1) функций, удовлетворяющих условию Липшица. На первом шаге выбирается любая из N точек равномерной сетки, определяющей в соответствии с теоремой 3.4 оптимальный алгоритм:

$$x_1 \in \left\{a+\frac{b-a}{N+1},\ a+2\frac{b-a}{N+1},\ \ldots,\ a+N\frac{b-a}{N+1}\right\}. \quad (3.34)$$

Пусть i информационных вычислений в точках $x_1, \ldots, x_i$ произведено, $x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания, $y_{ij}=f(x_{ij})$, $j=1, \ldots, i$, $x_{i0}=a$, $x_{i,\,i+1}=b$, $y_{i0}=y_0$, $y_{i,\,i+1}=y_{N+1}$. Пусть оставшиеся $N-i$ вычислений распределены так, что на отрезке $[x_{i,\,j-1}, x_{ij}]$ предписывается провести n_j вычислений, $j=1, \ldots, i+1$, и на каждом из отрезков применяется оптимальный алгоритм. Наилучший гарантированный результат при фиксированном числе n_j информационных вычислений для класса, соответствующего отрезку $[x_{i,\,j-1}, x_{ij}]$, с фиксированными значениями на обоих концах отрезка, согласно теореме 3.4, равен $\frac{(y_{ij}-y_{i,j-1})(x_{ij}-x_{i,j-1})}{2(n_j+1)}$, $j=1, \ldots, i+1$. Гарантированный результат в ситуации z^i при заданных $n_1, \ldots, n_{i+1}$ равен, очевидно, $\sum_{j=1}^{i+1}\frac{(y_{ij}-y_{i,j-1})(x_{ij}-x_{i,j-1})}{2(n_j+1)}$. Для выбора точки x_{i+1} необходимо решить задачу вида (3.9) отыскания величины

$$\varepsilon_N(z^i)=\min_{\substack{n_1,\ldots,n_{i+1}\in\{0,1,2,\ldots\}\\ \sum_{j=1}^{i+1} n_j=N-i}}\sum_{j=1}^{i+1}\frac{(y_{ij}-y_{i,j-1})(x_{ij}-x_{i,j-1})}{2(n_j+1)}.$$

Задачу можно решить с помощью описанных в п. 3 алгоритмов. После того как решение $n_1^i, \ldots, n_{i+1}^i$ найдено, за x_{i+1} можно принять любую из точек сетки, образованной объединением определенных в теореме 3.4 оптимальных наборов точек для классов, соответствующих отрезкам $[x_{i,j-1}, x_{ij}]$, $j = 1, \ldots, i+1$, т. е.

$$x_{i+1} \in \bigcup_{j=1}^{i+1} \left\{ x_{i,j-1} + k_j \frac{x_{ij} - x_{i,j-1}}{n_j^i + 1} \,\middle|\, k_j = 1, \ldots, n_j^i \right\},$$

$$i = 1, \ldots, N-1 \quad (3.35)$$

(если $n_j^i = 0$, то предполагается, что соответствующий набор пуст). Тем самым доказана следующая теорема.

Теорема 3.5. *Алгоритм* (3.34), (3.35) *последовательно-оптимален по точности на классе* (3.28). △

Оптимальный на один шаг алгоритм строится по той же схеме, что и для класса (3.1). Представим этот алгоритм в следующем виде:

$$x_1 = \frac{a+b}{2}, \quad x_{i+1} = \frac{x_{i,k-1} + x_{ik}}{2}, \quad i \geqslant 1, \quad (3.36)$$

где номер k определяется соотношением

$$(y_{ik} - y_{i,k-1})(x_{ik} - x_{i,k-1}) = \max_{j=1,\ldots,i+1} \{(y_{ij} - y_{i,j-1})(x_{ij} - x_{i,j-1})\}.$$

Нетрудно построить также и алгоритмы, последовательно-оптимальные по числу информационных вычислений.

7. О требованиях к модели вычислений, обеспечивающих возможность построения по единой схеме последовательно-оптимальных алгоритмов. Начнем с того, что укажем еще два функциональных класса, для которых последовательно-оптимальные алгоритмы можно построить по той же схеме, что и для классов (3.1), (3.28).

В [67] с помощью этой схемы последовательно-оптимальный алгоритм построен для класса неубывающих на отрезке $[a, b]$ функций, удовлетворяющих на этом отрезке условию Липшица с константой M.

В [177] последовательно-оптимальный алгоритм построен для класса функций, удовлетворяющих условию

$$-M_1(u-v) \leqslant f(u) - f(v) \leqslant M_2(u-v),$$

где $u, v \in [a, b]$, $u < v$, $M_1 > 0$, $M_2 > 0$.

Все перечисленные классы в совокупности с рассматриваемой в моделях вычислений информацией о функции обладают некоторыми общими особенностями. Это,

прежде всего, классы функций одной переменной. Информационные вычисления состоят в вычислении значений функции (и, быть может, ее производных) в точках области определения. Определяющее свойство модели, которое позволяет свести задачу построения последовательно-оптимального алгоритма к совокупности задач построения оптимальных алгоритмов, состоит в следующем.

Для любой реализуемой ситуации z^i, любых $f_1, f_2 \in F(z^i)$, $x_{i,j-1}$, x_{ij} ($x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания) должно быть

$$g(x) \stackrel{\text{def}}{=} \begin{cases} f_1(x), & x \in [x_{i,j-1}, x_{ij}], \\ f_2(x), & x \in K \setminus [x_{i,j-1}, x_{ij}] \end{cases} \in F(z^i). \quad (3.37)$$

Иными словами, варьирование функции из $F(z^i)$, не выводящее за пределы $F(z^i)$, на каждом из отрезков $[x_{i,j-1}, x_{ij}]$ можно производить независимо.

Сформулированное свойство позволяет построить при заданных вычислительных ресурсах оптимальный алгоритм для каждого из классов, соответствующих отрезкам $[x_{i,j-1}, x_{ij}]$, а затем, решив задачу оптимального распределения вычислительных ресурсов, построить последовательно-оптимальный алгоритм.

Условие (3.37) выполнено для многих моделей вычислений, список приведенных выше примеров можно продолжить. Так, это условие выполнено, если $F = W_\infty^r([a, b], M)$ (см. п. 4 § 2 гл. 1) и в модели принята эрмитова информация порядка r, т. е. информационному вычислению в точке x соответствует оператор

$$x(f) = (f(x), f'(x), \ldots, f^{(r-1)}(x)). \quad (3.38)$$

Таким образом, задача построения последовательно-оптимальных алгоритмов сводится к задачам построения оптимальных алгоритмов для классов $W_\infty^r([a, b], M)$ с фиксированной информацией (3.38) в одном или обоих концах отрезка интегрирования.

§ 4. Численный эксперимент

В данном параграфе, основанном на работах [72, 74], мы опишем некоторые особенности реализации алгоритмов из § 3. Основное внимание будет уделено численным экспериментам с алгоритмами и анализу полученных результатов. Важная роль численного эксперимента обу-

словлена тем обстоятельством, что зачастую на практике алгоритм, оптимальный в том или ином смысле на некотором функциональном классе F, приходится применять к функции $f \in F$, о которой априори известно больше, чем просто факт ее принадлежности к классу F. Например, о функции f может быть известно, что она монотонна, принимает заданные значения на концах отрезка, т. е. принадлежит к классу (3.28), и, кроме того, является кусочно-гладкой на данном отрезке. Теоретическое исследование и использование возможностей, предоставляемых такой дополнительной информацией, часто затруднены, так же как и теоретическая оценка эффективности алгоритмов, построенных для класса F, на определяемом дополнительной информацией подклассе класса F. Поэтому естественно прибегнуть к численному эксперименту. Кроме того, алгоритм, полученный в рамках некоторой модели вычислений, может быть модифицирован на основе тех или иных эвристических соображений для применения в условиях, не укладывающихся в исходную модель. Здесь при исследовании эффективности модифицированного алгоритма также не обойтись без численного эксперимента.

1. О конкретных реализациях алгоритмов интегрирования. Перечислим конкретные версии алгоритмов из § 3, для которых будет проведен сравнительный анализ эффективности.

Последовательно-оптимальные алгоритмы (ПОА), рассматриваемые ниже, реализованы в варианте, когда на каждом шаге из набора допустимых выбирается точка, ближайшая к точке, указываемой оптимальным на один шаг алгоритмом (ООША). Рассматриваются также ООША и «приближенный» последовательно-оптимальный алгоритм (ППОА). При выборе очередной точки ППОА задача (3.9) заменяется задачей (3.13), решение которой округляется до целочисленного. Перечисленные алгоритмы сравниваются с оптимальным пассивным алгоритмом, который для изучаемых в § 3 функциональных классов оптимален также и в множестве всех последовательных алгоритмов и именуется ниже оптимальным алгоритмом (ОА).

Все эти алгоритмы реализованы применительно к двум функциональным классам: классу (3.1), обозначаемому в этом параграфе через F_1, и классу (3.28), обозначаемому через F_2.

Далее, все алгоритмы реализованы в двух вариантах: варианте с фиксированным числом информационных вычислений и варианте с фиксированной точностью.

Наконец, при интегрировании функций из класса F_1 пользователь может не знать (или знать неточно) константу M. В модифицированных вариантах алгоритмов (МПОА, МООША, МППОА) константа M адаптивно оценивается в ходе вычислений способом, который будет описан ниже. Конечно, модификации уже не обладают свойствами оптимальности исходных алгоритмов.

2. Способы оценки константы Липшица. Опишем применяемый в модифицированных вариантах алгоритмов способ оценки константы M (этот способ предложен в [238]) и некоторые другие способы.

После i вычислений в точках $x_1, \ldots, x_i$ $(i \geqslant 2)$ текущее значение константы M в перечисленных в п. 1 модифицированных вариантах алгоритмов интегрирования полагалось равным

$$M_i = \sigma R_i, \tag{4.1}$$

где

$$R_i = \max_{\substack{1 \leqslant j,k \leqslant i \\ j \neq k}} \left| \frac{f(x_j) - f(x_k)}{x_j - x_k} \right|$$

— максимальная из абсолютных величин первых разделенных разностей, а σ — некоторый коэффициент «осторожности». Пусть $x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания, $x_{i,j-1} < x_{i+1} < x_{ij}$. Тогда

$$M_{i+1} = \max \left\{ M_i, \sigma \left| \frac{f(x_{i+1}) - f(x_{i,j-1})}{x_{i+1} - x_{i,j-1}} \right|, \right.$$
$$\left. \sigma \left| \frac{f(x_{i+1}) - f(x_{ij})}{x_{i+1} - x_{ij}} \right| \right\} \tag{4.2}$$

и, таким образом, вычисление оценок для константы Липшица не приводит к большим затратам машинного времени. Именно так проводилась оценка константы в модифицированных алгоритмах.

Способ (4.1) оценивания константы Липшица применим также и в многомерном случае, надо только в выражении для R_i заменить $|x_j - x_k|$ на $r(x_j, x_k)$, где r — расстояние. При этом, однако, формулой (4.2) воспользоваться уже нельзя, и на $(i+1)$-м шаге приходится вычислять не две, а i первых разделенных разностей. Ме-

тод (4.1) можно дополнить процедурой уточнения, проводимой через s шагов, т. е. на $(i+1)$-м шаге при $i = s, 2s, 3s, \ldots$ Процедура состоит в том, что после i шагов выбирается некоторая разделенная разность $(f(x_j) - f(x_k))/r(x_j, x_k)$, отрезок $[x_j, x_k]$ разбивается на несколько равных частей точками $u_1, \ldots, u_m$, вычисляются значения $f(u_1), \ldots, f(u_m)$ и за оценку M принимается величина

$$M_i = \sigma \max \left\{ R_i, \frac{|f(u_1) - f(x_j)|}{r(u_1, x_j)}, \frac{|f(u_2) - f(u_1)|}{r(u_2, u_1)}, \ldots \right.$$
$$\left. \ldots, \frac{|f(u_m) - f(u_{m-1})|}{r(u_m, u_{m-1})}, \frac{|f(x_k) - f(u_m)|}{r(x_k, u_m)} \right\}.$$

Номера j, k целесообразно выбирать из условия максимальности $|f(x_j) - f(x_k)|/r(x_j, x_k)$ среди модулей тех разделенных разностей, для которых описанная процедура уточнения еще не производилась.

Используя и иные эвристические приемы оценки константы M, применимые как в одномерном случае, так и в случае нескольких переменных. Так, в [153] предложено, наряду с величинами R_i, вычислять среднее приращение разделенной разности за i шагов $\Delta R_i = (R_i - R_2)/(i-2)$ и за оценку константы M после i шагов принимать в случае фиксированного N величину $M_i = R_i + (N-i)\Delta R_i$.

Можно вместо среднего арифметического ΔR_i приращений $R_j - R_{j-1}$, $j = 3, \ldots i$, вычислять их среднее геометрическое δR_i и за оценку M принимать после i шагов величину $M_i = \tau_i R_i (\delta R_i)^{N-i}$, где τ_i — некоторый поправочный коэффициент.

Еще один эвристический прием состоит в следующем. После i шагов строится некоторая аппроксимирующая функция $R^{(i)}$ такая, что $R^{(i)}(j) \approx R_j$, $j = 2, \ldots, i$, и за оценку M после i шагов принимается либо величина $\sigma R^{(i)}(N)$, либо, при соответствующем выборе $R^{(i)}$,— величина $\lim\limits_{t\to\infty} R^{(i)}(t)$. Можно выбирать, например, $R^{(i)}(t) = a_i/t + b_i$, вычисляя коэффициенты $a_i < 0$, b_i с помощью метода наименьших квадратов, либо по последним m значениям $R_{i-m+1}, \ldots, R_i$ (m — некоторое фиксированное число), либо по значениям $R_{\lfloor i/m \rfloor}, R_{\lfloor 2i/m \rfloor}, \ldots, R_i$. В этом случае за оценку константы M принимается величина b_i.

Допустимы различные модификации и комбинации описанных методов. Отметим лишь, что может оказаться

целесообразным применять такого рода приемы лишь на первой стадии вычислительного процесса — на первых N_1 шагах, не изменяя полученную оценку на последующих шагах.

Наконец, возможно использование априорных оценок константы M. Например, можно до начала информационных вычислений оценить норму градиента функции f.

Описанные приемы оценки константы Липшица применимы не только для задачи интегрирования, но и для других задач численного анализа.

3. О методике тестирования вычислительных алгоритмов. В настоящее время применяются различные методики тестирования вычислительных алгоритмов. Можно использовать наборы тестовых функций, либо составленные на основе тех или иных прикладных задач [332, 339, 341, 368], либо содержащие функции, которые традиционно считаются «трудными» для интегрирования [333].

Наш подход близок к подходу, развиваемому в работах [2, 11], где предлагается проводить тестирование на случайных выборках функций из некоторого семейства с последующим осреднением полученных результатов по выборке. Хотя рассматриваемые нами тестовые семейства функций не связаны непосредственно с конкретными прикладными вопросами, при их выборе учитываются особенности задач, часто встречающиеся на практике.

Тестирование алгоритмов было направлено на достижение следующих целей.

Во-первых, проверить работоспособность алгоритмов на достаточно представительных семействах тестовых функций.

Во-вторых, экспериментально исследовать эффективность алгоритмов на таких классах (тестовых семействах) функций, для которых не удается провести теоретическое исследование.

В-третьих, экспериментально проверить соответствие концепции последовательной оптимальности требованиям организации реальных вычислительных процессов. При этом сравнивались результаты применения ПОА, ППОА, ООПА и ОА.

Перейдем к описанию случайного механизма построения непрерывных кусочно гладких (с m изломами) функций, из которых составлялись тестовые серии. Зафиксируем набор гладких функций $f_1, \ldots, f_p$, заданных на отрезках $l_1, \ldots, l_p$ соответственно. Эти функции будем далее

называть базовыми. Процесс построения каждой тестовой функции разбивается на три этапа.

I. Внутри отрезка интегрирования $[a, b]=[0, 1]$ выбирается m случайных равновероятно распределенных точек $t_1, \ldots, t_m$, которые располагаются в порядке возрастания координат: $0=t_0<t_1<\ldots<t_m<t_{m+1}=1$.

II. Для каждого $i=0, 1, \ldots, m$ равновероятно среди чисел $1, \ldots, p$ выбирается номер базовой функции k_i, а затем случайная точка ξ_i в отрезке l_{k_i}.

III. Тестовая функция определяется следующим образом:

$$f(x)=d_i+f_{k_i}(\xi_i+x-t_i)-f_{k_i}(\xi_i), \quad x\in[t_i, t_{i+1}],$$

где i последовательно пробегает значения $0, 1, \ldots, m$, $d_0=0$, $d_i=f(t_i)$ при $i\geqslant 1$. При этом случайный выбор $\xi_i\in l_{k_i}$ осуществляется с таким расчетом, чтобы точка $\xi_i+t_{i+1}-t_i$ лежала в отрезке l_{k_i}.

При построении тестовых функций из класса F_1 в качестве базовых функций были выбраны три начальных отрезка ряда Фурье длины 4 со случайными коэффициентами:

$$\sum_{i=1}^{4}(a_k\cos kx+b_k\sin kx),$$

где a_k, b_k — независимые реализации равномерно распределенных на отрезке $[0, 1]$ случайных величин.

При построении тестовых функций из класса F_2 в качестве базовых были выбраны пять возрастающих функций: e^x на отрезке $[-5, 2]$, $\ln x$ на $[0.3, 5]$, x^2 на $[0, 2]$, $-1/x$ на $[0.5, 5]$, $\sin x$ на $[-\pi/2, \pi/2]$.

4. Результаты численных экспериментов. Будем обозначать оценочную точность (гарантированную после N вычислений точность $\varepsilon_N(z^N)$, определяемую формулой (4.15) гл. 1) через $\varepsilon_{\text{ОА}}$, $\varepsilon_{\text{ПОА}}$ и т. д. в зависимости от того, какой алгоритм используется.

В табл. 2 приводятся результаты численных экспериментов для класса F_2. Каждый результат получен осреднением соответствующей величины по 50 случайным испытаниям, т. е. по результатам применения соответствующих алгоритмов к 50 функциям тестовой серии случайно генерируемым описанным в п. 3 методом.

Первый показатель таблицы — отношение $\varepsilon_{\text{ПОА}}/\varepsilon_{\text{ОА}}$; второй — процент числа испытаний в серии, в которых ПОА дал более высокую точность, чем ОА; третий — про-

цент числа испытаний, в которых ПОА дал не менее высокую оценочную точность, чем ППОА; четвертый — отношение $\varepsilon_{\text{ПОА}}/\varepsilon_{\text{ООША}}$; пятый — процент числа испытаний, в которых $\varepsilon_{\text{ПОА}} < \varepsilon_{\text{ООША}}$.

Отметим, что фактическая точность интегрирования оказывается выше оценочной примерно па два порядка.

Таблица 2

Номер показателя	m	N			
		7	15	31	51
1	3	.88	.87	.84	.80
2		80	90	100	98
3		98	86	76	68
4		.99	.98	.97	.97
5		50	70	94	90
1	8	.84	.76	.70	.63
2		92	100	100	100
3		80	70	66	68
4		1.01	.99	.98	.98
5		34	60	74	84
1	20	.85	.70	.63	.62
2		92	100	100	100
3		88	64	62	52
4		1.02	1.01	1.00	.99
5		34	38	46	74

При тестировании алгоритмов, гарантирующих заданную точность, отмечены такие же тенденции, как и в случае фиксированного N. Эффект экономии числа узлов при

Таблица 3

ε	Класс					
	F_1			F_2		
0.02	13;	10;	0.77	18;	14;	0.82
0.01	25;	16;	0.62	36;	26;	0.74
0.005	50;	31;	0.61	72;	51;	0.73
2.5×10^{-4}	100;	59;	0.59	144;	97;	0.69
1.25×10^{-4}	200;	114;	0.57	288;	188;	0.67
6×10^{-5}	416;	235;	0.56	584;	379;	0.65

фиксированном ε от использования ПОА, ППОА по сравнению с ОА особенно нагляден. В табл. 3 приводятся показатели, характеризующие эффективность ППОА и ОА для различных значений ε при применении этих

алгоритмов к двум кусочно-гладким функциям (по одной функции из классов F_1, F_2), случайно генерируемым описанным в п. 3 методом. Для функции из F_1 выбрано значение $m = 3$; для функции из F_2 — значение $m = 20$.

В каждой клетке табл. 3 первое число — это количество информационных вычислений (узлов) ОА, гарантирующее достижение заданной точности; второе — число вычислений, которое потребовалось ППОА для достижения той же точности; третье число равно отношению второго к первому.

Более полные данные численных экспериментов можно найти в [74].

5. Выводы. На основании проведенных численных экспериментов, результаты которых частично приведены и прокомментированы в п. 4, можно сделать следующие общие выводы.

I. Все исследуемые алгоритмы зарекомендовали себя как достаточно эффективные и надежные методы численного интегрирования. Важной особенностью немодифицированных алгоритмов является способность установить гарантированную оценку для погрешности вычисления интеграла или вычислить интеграл с заданной погрешностью при условии, что имеющаяся информация об интегрируемой функции позволяет безошибочно определить принадлежность ее к одному из функциональных классов F_1, F_2 и тем самым выбрать соответствующий алгоритм интегрирования.

II. Для алгоритмов с фиксированным числом информационных вычислений (узлов) N с увеличением N проявляется и растет преимущество в точности ПОА над ОА и ООША (табл. 2). Дополнительные исследования, результаты которых не отражены в таблице, показали, что при достаточно большом числе узлов такое преимущество наблюдается в 100 % случаев. Та же тенденция четко просматривается для алгоритмов, обеспечивающих заданную точность ε. С уменьшением погрешности ε ярко проявляется, например, преимущество ППОА над ОА в числе узлов (табл. 3).

III. При не слишком больших значениях N ясно просматривается преимущество ПОА над ППОА (табл. 2). С ростом N это преимущество постепенно сходит на нет, что и понятно, так как решение целочисленной задачи (3.9) все точнее приближается решением ее непрерывного аналога (3.13).

Выводы II и III, как нам представляется, достаточно убедительно показывают на примере задачи численного интегрирования, что концепция последовательной оптимальности правильно и достаточно полно отражает специфику организации реальных вычислительных процессов. Так, огрубление задачи (3.9), фигурирующей в процессе построения ПОА, точнее, замена задачи (3.9) ее непрерывным аналогом (3.13) ведет к устойчивому возрастанию погрешности интегрирования. Показательным является также превосходство ПОА над ООША, особенно если вспомнить, что ПОА (так же, как и ППОА) реализованы в варианте, когда на каждом шаге из набора допустимых выбирается точка, ближайшая к точке, указываемой ООША.

Для модифицированных алгоритмов наблюдаются те же основные тенденции, что и для немодифицированных, но они выражены слабее. Оценочная точность модифицированных алгоритмов на одних и тех же функциях падает с увеличением коэффициента «осторожности» σ, что объясняется происходящим при этом завышением оценки для константы. Однако задание малого σ рискованно: так, в проведенных экспериментах при небольшом $(7 \div 15)$ числе узлов, фактическая точность МПОА при $\sigma = 1.1$ оказывалась хуже оценочной примерно в 10 % случаев, в то время как при $\sigma = 1.5$ — лишь в 3 %.

Подчеркнем в заключение, что ПОА целесообразно применять в первую очередь для интегрирования сложновычислимых функций, поскольку комбинаторная сложность (затраты на «внутренние потребности») ПОА существенно больше, чем у ОА. Кроме того, если число информационных вычислений N достаточно велико, то следует использовать ППОА, а не ПОА, так как комбинаторная сложность последнего выше.

§ 5. Оптимальное вычисление повторных интегралов

В этом параграфе, основанном на работе [442], рассматривается задача вычисления повторных интегралов от функций двух переменных, однако полученные результаты легко переносятся на случай произвольного числа переменных.

1. Задача повторного интегрирования. Рассмотрим класс F' функций, зависящих от двух переменных и удовлетворяющих по каждой из них условию Липшица:

$$F' = \{g \mid |g(x, u) - g(x', u)| \leqslant M|x - x'|,$$
$$|g(x, u) - g(x, u')| \leqslant P|u - u'|,\ x, u, x', u' \in [0, 1]\}.$$

В силу леммы 2.4 гл. 1 этот класс можно задавать с помощью некоторой квазиметрики. Если $g \in F'$, то функция $f(x) = \int\limits_0^1 g(x, u)\,du$ принадлежит классу

$$F = \{f \mid |f(x) - f(x')| \leqslant M|x - x'|,\ x, x' \in [0, 1]\},$$

поскольку

$$|f(x) - f(x')| \leqslant \int\limits_0^1 |g(x, u) - g(x', u)|\,du \leqslant M|x - x'|.$$

Если вся информация о функции g состоит в том, что $g \in F'$, то $f \in F$ — это вся информация о f. В самом деле, любая функция из F представима в виде интеграла от функции из F'.

Наша цель — построить оптимальные алгоритмы вычисления двойного интеграла $\iint\limits_{[0,1]^2} g(x, u)\,dx\,du$ как повторного, т. е. используя равенство

$$\iint\limits_{[0,1]^2} g(x, u)\,dx\,du = \int\limits_0^1 f(x)\,dx.$$

Важная особенность задачи заключается в том, что точное вычисление значений функции f, вообще говоря, невозможно. Алгоритмы оптимального вычисления значений $f(x) = \int\limits_0^1 g(x, u)\,du$ с любой наперед заданной точностью были построены в § 2, 3.

Отметим, что результаты данного параграфа применимы не только в случае, когда $f(x) = \int\limits_0^1 g(x, u)\,du$, но и в некоторых других случаях, когда необходимо найти интеграл от функции f, значения которой можно вычислить с любой наперед заданной точностью, например, в случае

$f(x) = \max\limits_u g(x, u)$ (вопросы оптимального вычисления значений таких функций рассматриваются в гл. 4).

Результаты данного параграфа непосредственно обобщаются на n-мерный случай. Действительно, построив алгоритм интегрирования функции двух переменных, можно вычислить интеграл от функции трех переменных по формуле

$$\iiint\limits_{[0,1]^3} h(x, u, t)\, dx\, du\, dt = \int\limits_0^1 f(x)\, dx,$$

где

$$f(x) = \iint\limits_{[0,1]^2} h(x, u, t)\, du\, dt,$$

и т. д.

2. Оптимальный пассивный алгоритм повторного интегрирования. Вопрос об оптимальном пассивном алгоритме повторного интегрирования сводится к вопросу об оптимальном интегрировании функции, значения которой вычисляются приближенно.

Пусть ν — число точек, в которых вычисляются значения функции f; $x_1, \ldots, x_\nu$ — сами эти точки, $0 \leqslant x_1 < x_2 < \ldots < x_\nu \leqslant 1$; $y_1, \ldots, y_\nu$ — полученные приближенные значения; $\delta_1, \ldots, \delta_\nu$ — погрешности вычислений, т. е.

$$|f(x_i) - y_i| \leqslant \delta_i, \quad i = 1, \ldots, \nu \tag{5.1}$$

(в этом случае говорят о вычислениях с неопределенными погрешностями). Используя (5.1) и условие Липшица, получаем для произвольного $x \in [0, 1]$

$$y_i - \delta_i - M|x - x_i| \leqslant f(x) \leqslant y_i + \delta_i + M|x - x_i|, \; i = 1, \ldots, \nu,$$

или, что то же самое,

$$\varphi_{1\nu}(x) \overset{\text{def}}{=} \max_{i=1,\ldots,\nu} \{y_i - \delta_i - M|x - x_i|\} \leqslant f(x) \leqslant$$
$$\leqslant \min_{i=1,\ldots,\nu} \{y_i + \delta_i + M|x - x_i|\} \overset{\text{def}}{=} \varphi_{2\nu}(x), \quad x \in [0, 1].$$

Нетрудно убедиться, что $\varphi_{1\nu}$ и $\varphi_{2\nu}$ являются соответственно точной минорантой и точной мажорантой [1]) в ситуации

[1]) Использование для точной миноранты и точной мажоранты тех же обозначений, что и в случае вычислений без погрешностей, не приведет в дальнейшем к каким-либо неясностям.

$w^\nu \overset{\text{def}}{=} (x^\nu, \delta^\nu, y^\nu)$ с $x^\nu = (x_1, \ldots, x_\nu)$, $y^\nu = (y_1, \ldots, y_\nu)$, $\delta^\nu = (\delta_1, \ldots, \delta_\nu)$, т. е. для функций из F, удовлетворяющих условию (5.1).

По аналогии со случаем точных вычислений (см. (3.4)) функция

$$\tilde{\beta}_*(w^\nu) \overset{\text{def}}{=} \int_0^1 \frac{1}{2}[\varphi_{1\nu}(x) + \varphi_{2\nu}(x)]\,dx =$$

$$= x_1 y_1 + \sum_{i=2}^{\nu} (x_i - x_{i-1}) \frac{y_i + y_{i+1}}{2} + (1 - x_\nu) y_\nu \qquad (5.2)$$

определяет центральную итоговую операцию. Будем считать, что величина (5.2) принимается в ситуации w^ν за приближенное значение интеграла. Максимальная возможная погрешность в ситуации w^ν есть

$$\varepsilon_\nu(w^\nu) \overset{\text{def}}{=} \int_0^1 [\varphi_{2\nu}(x) - \varphi_{1\nu}(x)]\,dx =$$

$$= \frac{1}{2}\left\{Mz_1^2 + 2\delta_1 z_1 + \sum_{i=2}^{\nu}\left[\frac{Mz_i^2}{2} - \frac{(y_i - y_{i-1})^2}{2M} - \frac{(\delta_i - \delta_{i-1})^2}{2M} + \right.\right.$$

$$\left.\left. + (\delta_i + \delta_{i-1}) z_i\right] + Mz_{\nu+1}^2 + 2\delta_\nu z_{\nu+1}\right\},$$

где

$$z_1 = x_1;\ z_i = x_i - x_{i-1},\ i = 2, \ldots, \nu;\ z_{\nu+1} = 1 - x_\nu.$$

Пусть N — общее число информационных вычислений, т. е. вычислений значений функции g; n_i — число значений g, используемых для вычисления приближенного значения y_i функции f в точке x_i, $i = 1, \ldots, \nu$. В силу (3.3) $\delta_i = P/(4n_i)$. Будем считать N, ν фиксированными, а $n_1, \ldots, n_\nu, x_1, \ldots, x_\nu$ — переменными величинами, подлежащими определению. Заменяя $n_1, \ldots, n_\nu, x_1, \ldots, x_\nu$ на $\delta_1, \ldots, \delta_\nu, z_1, \ldots, z_{\nu+1}$, ограничения на переменные можно записать в виде

$$\begin{gathered}
\sum_{i=1}^{\nu} 1/\delta_i = c, \quad \delta_i > 0, \qquad i = 1, \ldots, \nu, \\
\sum_{i=1}^{\nu+1} z_i = 1, \quad z_1 \geqslant 0, \quad z_{\nu+1} \geqslant 0, \\
|\delta_i - \delta_{i-1}| < Mz_i, \quad i = 2, \ldots, \nu.
\end{gathered} \qquad (5.3)$$

В первом равенстве $c \overset{\text{def}}{=} 4N/P$, т. е. первое ограничение есть попросту по-иному записанное условие $\sum_{i=1}^{\nu} n_i = N$. Последняя серия ограничений запрещает бесполезные вычисления, гарантируя, что каждое информационное вычисление дает новую информацию. Действительно, если, скажем, $\delta_i - \delta_{i-1} \geqslant Mz_i$ для некоторого $i \in \{2, \ldots, \nu\}$, то нетрудно видеть, что в наихудшем случае $y_{i-1} = y_i$ вычисление значения функции f в точке x_i не дает новой информации.

Отметим, что в ограничениях (5.3) и в последующих построениях мы не учитываем целочисленности $n_1, \ldots, n_\nu$.

Из неравенств (5.1) и того факта, что $f \in F$, следует, что вектор y^ν должен удовлетворять ограничениям

$$|y_i - y_{i-1}| \leqslant Mz_i + \delta_{i-1} + \delta_i, \quad i = 2, \ldots, \nu. \tag{5.4}$$

Следующее определение естественным образом распространяет определение оптимальности из п. 1 § 4 гл. 1 (см. также теорему 4.3 гл. 1) на случай приближенных вычислений.

Вектор (x_0^ν, δ_0^ν) (в совокупности с итоговой операцией (5.2)) назовем *оптимальным по точности на классе F' пассивным алгоритмом повторного интегрирования при фиксированных N, ν*, если

$$\sup_{y^\nu} \varepsilon_\nu (x_0^\nu, \delta_0^\nu, y^\nu) = \min_{x^\nu, \delta^\nu} \sup_{y^\nu} \varepsilon_\nu (x^\nu, \delta^\nu, y^\nu),$$

где x^ν, δ^ν, y^ν удовлетворяют условиям (5.3), (5.4).

В дальнейшем мы откажемся от предположения о том, что ν фиксировано, и найдем оптимальное значение ν_0.

Ясно, что

$$\sup_{y^\nu} \varepsilon_\nu (x^\nu, \delta^\nu, y^\nu) = \varepsilon_\nu (x^\nu, \delta^\nu, (0, \ldots, 0)) \overset{\text{def}}{=} \omega (x^\nu, \delta^\nu) =$$

$$= \frac{1}{2} \left\{ Mz_1^2 + 2\delta_1 z_1 + \sum_{i=2}^{\nu} \left[\frac{Mz_i^2}{2} - \frac{(\delta_i - \delta_{i-1})^2}{2M} + \right.\right.$$

$$\left.\left. + (\delta_i + \delta_{i-1}) z_i \right] + Mz_{\nu+1}^2 + 2\delta_\nu z_{\nu+1} \right\},$$

и, таким образом, задача состоит в отыскании величины $\min_{x^\nu, \delta^\nu} \omega(x^\nu, \delta^\nu)$, а также x_0^ν, δ_0^ν, при которых минимум достигается.

Отбросим временно ограничения

$$z_1 \geqslant 0,\ z_{\nu+1} \geqslant 0,\ |\delta_i - \delta_{i-1}| < Mz_i,\quad i = 2,\ \ldots,\ \nu.$$

Легко видеть, что

$$\min_{x^\nu} \omega\left(x^\nu, \delta^\nu\right) = \min_{(z_1,\ldots,z_{\nu+1})\in S_{\nu+1}} \frac{1}{2}\left\{Mz_1^2 + 2\delta_1 z_1 + \right.$$

$$+ \sum_{i=2}^{\nu}\left[\frac{Mz_i^2}{2} - \frac{(\delta_i - \delta_{i-1})^2}{2M} + (\delta_i + \delta_{i-1})\, z_i\right] + Mz_{\nu+1}^2 +$$

$$\left. + 2\delta_\nu z_{\nu+1}\right\} = \frac{1}{4\nu M}\left[\left(M + 2\sum_{i=1}^{\nu}\delta_i\right)^2 - 4\nu\sum_{i=1}^{\nu}\delta_i^2\right],$$

где

$$S_{\nu+1} = \left\{(z_1,\ \ldots,\ z_{\nu+1})\,\middle|\,\sum_{i=1}^{\nu+1} z_i = 1\right\},$$

причем минимум достигается при

$$z_1^0 = \frac{1}{2\nu} + \frac{\delta - \delta_1}{M},\ z_i^0 = \frac{1}{\nu} + \frac{2\delta - \delta_i - \delta_{i-1}}{M},\quad i = 2,\ \ldots,\ \nu,$$

$$z_{\nu+1}^0 = \frac{1}{2\nu} + \frac{\delta - \delta_\nu}{M};\qquad \delta \overset{\text{def}}{=} \sum_{i=1}^{\nu}\frac{\delta_i}{\nu}.$$

Следовательно, $\min\limits_{x^\nu} \omega\left(x^\nu,\ \delta^\nu\right)$ достигается при $x_0^\nu = \left(x_1^0, \ldots, x_\nu^0\right)$ с $x_i^0 = \sum\limits_{j=1}^{i} z_j^0$, $i = 1, \ldots, \nu$. Имеем

$$\min_{x^\nu,\delta^\nu} \omega\left(x^\nu, \delta^\nu\right) = \frac{1}{4\nu M}\min_{\delta \geqslant \nu/c}\ \min_{\delta_1,\ldots,\delta_\nu > 0}\left[(M + 2\nu\delta)^2 - 4\nu\sum_{i=1}^{\nu}\delta_i^2\right], \tag{5.5}$$

где

$$\sum_{i=1}^{\nu}\frac{1}{\delta_i} = c,\quad \sum_{i=1}^{\nu}\delta_i = \nu\delta$$

(ограничения $z_1 \geqslant 0$, $z_{\nu+1} \geqslant 0$, $|\delta_i - \delta_{i-1}| < Mz_i$, $i = 2, \ldots, \nu$, временно отброшены). Неравенство $\delta \geqslant \nu/c$ включено в число ограничений по той причине, что для сов-

местимости ограничений $\sum_{i=1}^{\nu} \frac{1}{\delta_i} = c$, $\delta_1 > 0, \ldots, \delta_\nu > 0$ и $\sum_{i=1}^{\nu} \delta_i = \nu\delta$ необходимо и достаточно, чтобы выполнялось неравенство

$$\min_{\substack{\delta_1,\ldots,\delta_\nu>0 \\ \sum_{i=1}^{\nu} 1/\delta_i = c}} \sum_{i=1}^{\nu} \delta_i = \frac{\nu^2}{c} \leqslant \nu\delta.$$

Для отыскания (5.5) рассмотрим сначала задачу отыскания

$$\max_{\delta_1,\ldots,\delta_\nu>0} \sum_{i=1}^{\nu} \delta_i^2 \text{ при } \sum_{i=1}^{\nu} \frac{1}{\delta_i} = c, \quad \sum_{i=1}^{\nu} \delta_i = \nu\delta. \tag{5.6}$$

Выпишем для задачи (5.6) функцию Лагранжа и ее первые два дифференциала:

$$L = \sum_{i=1}^{\nu} \delta_i^2 - \lambda \sum_{i=1}^{\nu} \frac{1}{\delta_i} - \mu \sum_{i=1}^{\nu} \delta_i,$$

$$dL = \sum_{i=1}^{\nu} \left(2\delta_i + \frac{\lambda}{\delta_i^2} - \mu\right) d\delta_i, \quad d^2L = \sum_{i=1}^{\nu} \left(2 - \frac{2\lambda}{\delta_i^3}\right) (d\delta_i)^2.$$

Точка $\delta_0^\nu = (\delta_1^0, \ldots, \delta_\nu^0)$, доставляющая максимум в (5.6), должна быть решением системы

$$2\delta_i^3 - \mu\delta_i^2 = -\lambda, \quad i = 1, \ldots, \nu, \quad \sum_{i=1}^{\nu} \frac{1}{\delta_i} = c, \quad \sum_{i=1}^{\nu} \delta_i = \nu\delta.$$

Уравнение $2x^3 - \mu x^2 = -\lambda$ имеет не более двух положительных корней α и β, $0 < \alpha \leqslant \beta$. Пусть $\delta_1^0 = \ldots = \delta_k^0 = \alpha$, $\delta_{k+1}^0 = \ldots = \delta_\nu^0 = \beta$. Чтобы установить, является ли δ_0^ν точкой максимума, рассмотрим квадратичную форму

$$d^2L = \left(2 - \frac{2\lambda}{\alpha^3}\right) \sum_{i=1}^{k} (d\delta_i)^2 + \left(2 - \frac{2\lambda}{\beta^3}\right) \sum_{i=k+1}^{\nu} (d\delta_i)^2$$

при

$$\sum_{i=1}^{\nu} d\delta_i = 0, \quad \sum_{i=1}^{k} \frac{d\delta_i}{\alpha^2} + \sum_{i=k+1}^{\nu} \frac{d\delta_i}{\beta^2} = 0.$$

Последние равенства могут выполняться лишь в случае, если выполнено одно из двух условий

$$\begin{aligned} &\sum_{i=1}^{\nu} d\delta_i = 0, \quad \alpha = \beta; \\ &\sum_{i=1}^{k} d\delta_i = 0, \quad \sum_{i=k+1}^{\nu} d\delta_i = 0, \quad \alpha < \beta. \end{aligned} \tag{5.7}$$

Учитывая, что $0 < \alpha \leqslant \beta$ и α, β — корни уравнения $2x^3 - \mu x^2 = -\lambda$, получаем

$$\lambda = \frac{2\alpha^2\beta^2}{\alpha+\beta}, \quad 2 - \frac{2\lambda}{\alpha^3} = 2\,\frac{\alpha(\alpha+\beta) - 2\beta^2}{\alpha(\alpha+\beta)} \leqslant 0,$$

$$2 - \frac{2\lambda}{\beta^3} = 2\,\frac{\beta(\alpha+\beta) - 2\alpha^2}{\beta(\alpha+\beta)} \geqslant 0.$$

Поэтому для того, чтобы в точке δ_0^ν достигался максимум, необходимо выполнение одного из двух условий: $\alpha = \beta$ или $\alpha < \beta$, $k = \nu - 1$ (в противном случае квадратичная форма d^2L при соблюдении любого из условий (5.7) была бы положительно определенной или неопределенной и, следовательно, точка δ_0^ν не могла бы быть точкой максимума). В обоих случаях необходимые условия максимума имеют вид

$$\frac{\nu-1}{\alpha} + \frac{1}{\beta} = c, \quad (\nu-1)\alpha + \beta = \nu\delta, \quad 0 < \alpha \leqslant \beta.$$

Полагая $\alpha = \delta - \gamma$, $\beta = \delta + (\nu - 1)\gamma$, $\gamma \geqslant 0$, убеждаемся, что

$$\gamma(\delta) \overset{\text{def}}{=} \frac{1}{2c(\nu-1)}\Big[(\nu-2)(c\delta-\nu) + \\ + \sqrt{(\nu-2)^2(c\delta-\nu)^2 + 4c\delta(\nu-1)(c\delta-\nu)}\Big]$$

есть положительный корень уравнения $\frac{\nu-1}{\delta-\gamma} + \frac{1}{\delta+(\nu-1)\gamma} = 0$.

Таким образом, задача (5.5) сводится к вычислению величины

$$\min_{\delta \geqslant \nu/c} \psi(\delta), \text{ где } \psi(\delta) = \frac{M}{4\delta} + \frac{M\delta - \nu(\nu-1)\gamma^2(\delta)}{M}.$$

Нетрудно проверить, что $\psi''(\delta) < 0$ при $\delta \geqslant \nu/c$, поэтому минимум достигается при $\delta = \nu/c = \alpha = \beta$ или $\delta = \infty$.

В первом случае $\delta_1^0 = \ldots = \delta_\nu^0 = \delta$ и

$$\min_{x^\nu, \delta^\nu} \omega(x^\nu, \delta^\nu) = \frac{M}{4\nu} + \frac{\nu}{c}.$$

Во втором случае минимум функции $\omega(x^\nu, \delta^\nu)$ по x^ν, δ^ν, удовлетворяющим условиям (5.3), достигается на границе множества (5.3), т. е. либо $z_1 = 0$, либо $Mz_i = |\delta_i - \delta_{i-1}|$ при некотором $i \in \{2, \ldots, \nu\}$, либо $z_{\nu+1} = 0$. В п. 4 мы докажем, что в данном случае обязательно $Mz_i = |\delta_i - \delta_{i-1}|$ при некотором $i \in \{2, \ldots, \nu\}$. Это означает, что соответствующее вычисление значения функции f не является необходимым, так как оно в наихудшем случае не дает новой информации. В этом случае, не меняя общего числа информационных вычислений, будем считать, что число точек в которых вычисляются значения f, не больше ν. Формально можно считать, что в некоторых из этих ν точек допустимы вычисления с бесконечной ошибкой. Тем самым мы несколько уточняем данное выше определение оптимальности, допуская бесконечные значения $\delta_1, \ldots, \delta_\nu$.

Объединяя оба случая, получаем

$$\min_{x^\nu, \delta^\nu} \omega(x^\nu, \delta^\nu) = \min_{j \leqslant \nu} \left(\frac{M}{4j} + \frac{j}{c}\right).$$

Легко убедиться, что минимум в правой части достигается при $j = j(\nu)$, где

$$j(\nu) = \max\{j \mid j \leqslant \nu,\ j(j-1) \leqslant Mc/4 = NM/P\}.$$

Заметим, что последнее неравенство эквивалентно неравенству

$$M/(4j) + j/c \leqslant M/[4(j-1)] + (j-1)/c.$$

Заменяя в полученных ранее формулах для x_0^ν величины δ^ν на δ_0^ν, а также заменяя ν на $j(\nu)$, получим, что оптимальный алгоритм $\left(x_0^{j(\nu)}, \delta_0^{j(\nu)}\right)$ задается соотноше-

ниями

$$x_1^0 = \frac{1}{2j(\nu)}, \quad x_2^0 = \frac{3}{2j(\nu)}, \ldots, x_{j(\nu)}^0 = \frac{2j(\nu)-1}{2j(\nu)},$$

$$\delta_1^0 = \ldots = \delta_{j(\nu)}^0 = \frac{j(\nu)}{c} = \frac{j(\nu)P}{4N}.$$

В самом деле, отбрасывание ограничений

$$z_1 \geqslant 0, \ z_{\nu+1} \geqslant 0, \ |\delta_i - \delta_{i-1}| < Mz_i, \quad i = 2, \ldots, \nu,$$

в исходной задаче не повлияло на результат, так как полученная точка минимума удовлетворяет всем ограничениям исходной задачи.

В построении алгоритма $\left(x_0^{j(\nu)}, \delta_0^{j(\nu)}\right)$ и состоит основной результат данного параграфа.

Если фиксируя по-прежнему число N информационных вычислений, вообще отказаться от фиксации ν, то из проведенных построений следует, что оптимальное число ν_0 вычислений значений функции определяется соотношением

$$\nu_0 = \max\{\nu \mid \nu(\nu-1) \leqslant NM/P\}. \tag{5.8}$$

В этом случае оптимальный алгоритм задан формулами

$$x_1^0 = \frac{1}{2\nu_0}, \quad x_2^0 = \frac{3}{2\nu_0}, \ldots, x_{\nu_0}^0 = \frac{2\nu_0 - 1}{2\nu_0},$$

$$\delta_1^0 = \ldots = \delta_{\nu_0}^0 = \frac{\nu_0}{c} = \frac{\nu_0 P}{4N}.$$

Еще раз обратим внимание на следующее. Во-первых, говоря об оптимальных алгоритмах, мы имеем в виду оптимальность в классе алгоритмов повторного интегрирования. Во-вторых, в определении оптимального алгоритма повторного интегрирования не учитывается дискретность множеств возможных значений для $\delta_1, \ldots, \delta_\nu$ (иными словами, целочисленность $n_1, \ldots, n_\nu$).

3. Аналог последовательно-оптмального алгоритма для задачи повторного интегрирования. В этом пункте мы покажем, как результаты § 3 по построению последовательно-оптимальных алгоритмов переносятся на случай вычисления повторных интегралов. При этом будем считать, что точность ε, которую нужно обеспечить в результате вычислений фиксирована.

Найдем сначала число информационных вычислений N, гарантирующее при использовании оптимального пас-

сивного алгоритма повторного интегрирования (в случае нефиксированного ν) достижение точности ε. Учитывая целочисленность ν, следует исходить из (5.8). Мы, однако, пренебрежем целочисленностью ν и N. Тогда оптимальное ν_0 находится из условия

$$\frac{M}{4\nu_0}+\frac{\nu_0}{c}=\min_{j>0}\left(\frac{M}{4j}+\frac{j}{c}\right),$$

т. е. $\nu_0=(NM/P)^{1/2}$, а N — из условия

$$\min_{x^{\nu_0},\delta^{\nu_0}}\omega\left(x^{\nu_0},\delta^{\nu_0}\right)=\frac{M}{4\nu_0}+\frac{\nu_0}{c}=\frac{\sqrt{MP}}{2\sqrt{N}}=\varepsilon,$$

т. е. $N=MP/(\varepsilon)^2$, $\nu_0=M/(2\varepsilon)$. Следовательно, оптимальной точностью, которую следует задавать при вычислении значений функции $f(x)=\int\limits_0^1 g(x,u)\,du$, является $\delta_0=\nu_0/c=\varepsilon/2$. Зафиксируем эту точность при вычислении всех значений функции f.

Перейдем непосредственно к построению аналога последовательно-оптимального алгоритма из § 3. Первое информационное вычисление производится в точке

$$x_1\in\left\{\frac{1}{2\nu_0},\frac{3}{2\nu_0},\ldots,\frac{2\nu_0-1}{2\nu_0}\right\}.$$

Пусть произведено i вычислений в точках $x_1,\ldots,x_i$, пусть $x_{i1},\ldots,x_{ii}$ — перестановка чисел $x_1,\ldots,x_i$ в порядке возрастания, $y_{ij}=f(x_{ij})$, $j=1,\ldots,i$. Обозначим через p минимальное число оставшихся вычислений значений функции f, при котором точность ε еще может быть достигнута. Пусть n_1 из p вычислений предстоит произвести на $[0,x_{i1})$, n_2 — на (x_{i1},x_{i2}), $\ldots$, n_i — на $(x_{i,i-1},x_{ii})$, n_{i+1} — на $(x_{ii},1]$, $\sum\limits_{j=1}^{i+1}n_j=p$. Числа p, $n_1,\ldots,n_{i+1}$ будут определены позднее.

Рассмотрим произвольный из интервалов $(x_{i,j-1},x_{ij})$, $j=2,\ldots,i$. Мы хотим определить оптимальные точки

$$t_1,\ldots,t_{n_j},\ t_0\overset{\text{def}}{=}x_{i,j-1}<t_1<t_2<\ldots<t_{n_j}<x_{ij}\overset{\text{def}}{=}t_{n_j+1},$$

в которых следует вычислить значения функции f, т. е.

точки, реализующие минимум в выражении

$$\min_{t_1,\ldots,t_{n_j}} \max_{v_1,\ldots,v_{n_j}} \sum_{k=1}^{n_j+1} \left[\frac{M(t_k - t_{k-1})^2}{4} - \frac{(v_k - v_{k-1})^2}{4M} + \right.$$

$$\left. + \delta_0 (t_k - t_{k-1}) \right] \stackrel{\text{def}}{=} \omega_j, \quad (5.9)$$

где $v_k = f(t_k)$, $k = 0, 1, \ldots, n_j + 1$, $\delta_0 = \varepsilon/2$ — заданная точность вычисления значений функции f. Функция под знаком минимакса в (5.9) получается как частный случай из выражения для $\varepsilon_\nu(w^\nu)$. Ограничения на переменные в (5.9) выписываются по аналогии с (5.4):

$$|v_k - v_{k-1}| \leqslant M(t_k - t_{k-1}) + 2\delta_0, \quad k = 1, \ldots, n_j + 1.$$

Легко убедиться в том, что

$$t_k = x_{i,j-1} + k \frac{x_{ij} - x_{i,j-1}}{n_j + 1},$$

$$\omega_j = \frac{M(x_{ij} - x_{i,j-1})^2}{4(n_j + 1)} - \frac{(y_{ij} - y_{i,j-1})^2}{4M(n_j + 1)} + \delta_0 (x_{ij} - x_{i,j-1}).$$

Рассматривая таким же образом промежутки $[0, x_{i1})$, $(x_{ii}, 1]$, приходим к заключению, что для отыскания оптимальных $n_1, \ldots, n_{i+1}$ необходимо решить ту же задачу целочисленного программирования, что и в случае, когда значения функции f вычисляются точно (см. (3.7), (3.9)). Именно, следует найти $n_1, \ldots, n_{i+1}$, реализующие минимум в выражении

$$\min_{\substack{n_1,\ldots,n_{i+1} \in \{0,1,2,\ldots\} \\ \sum_{j=1}^{i+1} n_j = p}} \left[\frac{M}{4} \cdot \frac{x_{i1}^2}{n_1 + 1/2} + \right.$$

$$+ \frac{1}{4M} \sum_{j=2}^{i} \frac{(x_{ij} - x_{i,j-1})^2 M^2 - (y_{ij} - y_{i,j-1})^2}{n_j + 1} +$$

$$\left. + \frac{M}{4} \cdot \frac{(1 - x_{ii})^2}{n_{i+1} + 1/2} + \delta_0 \right] \stackrel{\text{def}}{=} \omega(p).$$

После того как решение найдено, за x_{i+1} следует принять любую из точек $t_1, \ldots, t_{n_j}$, реализующих минимум в одной из задач (5.9), $j = 2, \ldots, i$, или любую из точек, реализующих минимум в аналогичных задачах для $[0, x_{i+1})$,

$(x_{ii}, 1]$. При этом p — это минимальное целое число, для которого $\omega(p) \leqslant \varepsilon$.

Подчеркнем, что мы зафиксировали одну и ту же точность δ_0 для всех вычислений значений функции f. Можно было бы при определении x_{i+1} рассматривать погрешности вычисления значений как параметры, подлежащие оптимизации (в пределах каждого из промежутков $[0, x_{i1})$, (x_{i1}, x_{i2}), ..., $(x_{ii}, 1]$, как нетрудно показать, все вычисления следует производить с одной и той же для данного промежутка погрешностью). Без особого труда можно было бы сформулировать задачи отыскания оптимальных погрешностей. Решение этих задач, однако, сильно увеличило бы комбинаторную сложность алгоритма вычисления повторного интеграла, что сделало бы применение такого алгоритма нецелесообразным. Изложенный подход позволяет решать задачу многомерного интегрирования фактически с помощью алгоритма интегрирования функций одной переменной.

4. Вспомогательное утверждение. В этом пункте мы дадим доказательство вспомогательного утверждения, использованного в построениях п. 2. Введем некоторые дополнительные обозначения (сохраняя при этом все обозначения п. 2). Зафиксировав произвольное δ^ν, удовлетворяющее ограничениям

$$\sum_{i=1}^{\nu} \frac{1}{\delta_i} = c, \quad \delta_i > 0, \quad i = 1, \ldots, \nu,$$

положим

$$z = (z_1, \ldots, z_{\nu+1}), \ \omega(z) = \omega(x^\nu, \delta^\nu),$$
$$A = \{z \in S_{\nu+1} \mid z_i \geqslant a_i, \quad i = 1, \ldots, \nu + 1\},$$
$$A_i = \{z \in A \mid z_i = a_i\}, \quad \partial A = \bigcup_{i=1}^{\nu+1} A_i,$$

где

$$S_{\nu+1} = \left\{ z \,\middle|\, \sum_{i=1}^{\nu+1} z_i = 1 \right\},$$
$$a_1 = a_{\nu+1} = 0, \quad a_i = \frac{|\delta_i - \delta_{i-1}|}{M}, \quad i = 2, \ldots, \nu.$$

Сформулируем доказываемое утверждение в виде следующей леммы.

Лемма 5.1. *Если*

$$\min_{z\in A}\omega(z)=\omega(z^*),\quad z^*\in\partial A,$$

то

$$z^*\in\bigcup_{i=2}^{\nu}A_i.$$

Доказательство. Как было показано в п. 2, $\min\limits_{z\in S_{\nu+1}}\omega(z)=\omega(z^0)$, где

$$z_1^0=\frac{1}{2\nu}+\frac{\delta-\delta_1}{M},\quad z_i^0=\frac{1}{\nu}+\frac{2\delta-\delta_i-\delta_{i-1}}{M},\quad i=2,\ldots,\nu,$$

$$z_{\nu+1}^0=\frac{1}{2\nu}+\frac{\delta-\delta_\nu}{M},\quad \delta=\sum_{i=1}^{\nu}\frac{\delta_i}{\nu}.$$

Очевидно,

$$\begin{aligned}
&z_1^0>a_1\Leftrightarrow M+2\nu\delta>2\nu\delta_1,\\
&z_i^0>a_i\Leftrightarrow M+2\nu\delta>2\nu\max\{\delta_{i-1},\delta_i\},\quad i=2,\ldots,\nu,\\
&z_{\nu+1}^0>a_{\nu+1}\Leftrightarrow M+2\nu\delta>2\nu\delta_\nu.
\end{aligned}\tag{5.10}$$

Положив $I_0=\{i\,|\,z_i^0\leqslant a_i\}$, покажем, что

$$z^*\in\bigcup_{i\in I_0}A_i.\tag{5.11}$$

Действительно, в противном случае

$$z_i^0>a_i\ \text{при всех}\ i\in\{i\,|\,z_i^*=a_i\}.\tag{5.12}$$

Далее, легко видеть, что в силу строгой выпуклости по z функции ω выпуклое множество A не должно иметь внутренней относительно $S_{\nu+1}$ точки, принадлежащей множеству

$$B=\{z\in S_{\nu+1}\,|\,\omega(z)\leqslant\omega(z^*)\}.$$

Приходим к противоречию, поскольку $z^*+\lambda(z^0-z^*)\in B$ при $0\leqslant\lambda\leqslant1$ и в то же время точка $z^*+\lambda(z^0-z^*)$ принадлежит вследствие (5.12) относительной внутренности множества A при достаточно малом $\lambda>0$. Тем самым соотношение (5.11) доказано.

Рассматривая задачу вычисления

$$\omega(\tilde z)=\min_{z\in S_{\nu+1},z_1=0}\omega(z),\quad \tilde z=(0,\tilde z_2,\ldots,\tilde z_{\nu+1})\in S_{\nu+1},$$

по аналогии с (5.10) имеем

$$\tilde{z}_i > a_i \Leftrightarrow M + 2\nu\delta > (2\nu - 1)\max\{\delta_{i-1}, \delta_i\} + \delta_1, \quad i = 2, \ldots, \nu, \tag{5.13}$$
$$\tilde{z}_{\nu+1} > a_{\nu+1} \Leftrightarrow M + 2\nu\delta > (2\nu - 1)\delta_\nu + \delta_1.$$

Если $z^* \in A_1$, то тем же способом, что и для (5.11), можно убедиться в справедливости соотношения

$$z^* \in \bigcup_{i \in \tilde{I}} A_i, \tag{5.14}$$

где $\tilde{I} = \{i > 1 \mid \tilde{z}_i \leqslant a_i\}$.

Предположим, что $z^* \in A_1 \setminus \left(\bigcup_{i=2}^{\nu+1} A_i\right)$. Тогда, в силу (5.11), $z_1^0 \leqslant a_1$ и, в силу (5.10), $M + 2\nu\delta \leqslant 2\nu\delta_1$. С другой стороны, в этом случае $z^* = \tilde{z}$ и в соответствии с (5.13) при $i = 2$ имеем $M + 2\nu\delta > 2\nu\delta_1$. Полученное противоречие показывает, что $z^* \notin A_1 \setminus \left(\bigcup_{i=2}^{\nu+1} A_i\right)$. Точно так же убеждаемся, что $z^* \notin A_{\nu+1} \setminus \left(\bigcup_{i=1}^{\nu} A_i\right)$.

Рассматривая задачу вычисления

$$\omega(\hat{z}) = \min_{z \in S_{\nu+1}, z_1 = z_{\nu+1} = 0} \omega(z), \quad \hat{z} = (0, \hat{z}_2, \ldots, \hat{z}_\nu, 0) \in S_{\nu+1},$$

по аналогии с (5.10), (5.13) имеем

$$\hat{z}_i > a_i \Leftrightarrow M + 2\nu\delta > 2(\nu - 1)\max\{\delta_{i-1}, \delta_i\} + \delta_1 + \delta_\nu,$$
$$i = 2, \ldots, \nu. \tag{5.15}$$

Предположим, что $z^* \in (A_1 \cap A_{\nu+1}) \setminus \left(\bigcup_{i=2}^{\nu} A_i\right)$. Тогда, в силу (5.14), $\tilde{z}_{\nu+1} \leqslant a_{\nu+1}$ и, в силу (5.13), $M + 2\nu\delta \leqslant (2\nu - 1)\delta_\nu + \delta_1$. С другой стороны, в этом случае $z^* = \hat{z}$ и в соответствии с (5.15) при $i = \nu$ имеем $M + 2\nu\delta > (2\nu - 1)\delta_\nu + \delta_1$. Полученное противоречие показывает, что $z^* \notin (A_1 \cap A_{\nu+1}) \setminus \left(\bigcup_{i=2}^{\nu} A_i\right)$. Итак, единственная оставшаяся возможность состоит в том, что $z^* \in \bigcup_{i=2}^{\nu} A_i$. $\triangle$

§ 6. Вычисление кратных интегралов с помощью разверток типа кривой Пеано

Рассмотрим задачу вычисления интеграла от функции f по n-мерному единичному кубу

$$K=\{u=(u^1, \ldots, u^n) \mid -1/2 \leqslant u^i \leqslant 1/2,\ i=1, \ldots, n\}.$$

Пусть φ: $[0, 1] \to K$ — однозначное непрерывное отображение отрезка $[0, 1]$ на K. Тогда при определенных условиях на φ вычисление интеграла $\int\limits_K f(u)\,du$ можно точно или приближенно заменить вычислением интеграла $\int\limits_0^1 f(\varphi(x))\,dx$. Такая идея в случае, когда φ — отображение Пеано (см., например, [156]), высказана в [51]. Аналогичный способ редукции размерности при решении экстремальных задач был применен в [238].

В данном параграфе будет построено отображение φ_m, приближающее с точностью не хуже 2^{-m} по каждой координате отображение Пеано φ, и показано, что при достаточно большом m величина $\left|\int\limits_K f(u)\,du - \int\limits_0^1 f(\varphi_m(x))\,dx\right|$ сколь угодно мала. Способ построения отображения φ_m отличается от способа [238] лишь в деталях. Однако структура отображения φ_m существенно используется при доказательстве основной в данном параграфе теоремы 6.1, поэтому нам придется воспроизвести здесь некоторые из построений [238].

1. Определение отображений φ и φ_m. На первом этапе куб разделяется координатными плоскостями на 2^n кубов «первого разбиения» (с длиной ребра, равной 1/2). Эти кубы нумеруются числами z_1 от 0 до 2^n-1, причем куб первого разбиения с номером z_1 обозначается через $K(z_1)$. Каждый из кубов первого разбиения в свою очередь разделяется на 2^n кубов второго разбиения (с длиной ребра, равной 1/4) плоскостями, которые параллельны координатным плоскостям и проходят через середины ребер кубов первого разбиения. При этом кубы второго разбиения, входящие в куб $K(z_1)$, нумеруются числами z_2 от 0 до 2^n-1, причем куб второго разбиения с номером z_2, входящий в $K(z_1)$, обозначается через $K(z_1, z_2)$.

Продолжая этот процесс, строим кубы любого m-го разбиения с длиной ребра, равной 2^{-m}. Будем обозначать

кубы m-го разбиения через $K(z_1, \ldots, z_m)$, при этом

$$K(z_1) \supset K(z_1, z_2) \supset \ldots \supset K(z_1, z_2, \ldots, z_m), \quad 0 \leqslant z_j \leqslant 2^n - 1, \quad j = 1, \ldots, m.$$

Осуществим теперь деление отрезка [0, 1] на 2^n равных частей, каждую из которых в свою очередь разделим на 2^n равных частей, и т. д., причем элементы каждого разбиения нумеруются слева направо числами z_j (j — номер разбиения) от 0 до $2^n - 1$. При этом промежутки m-го разбиения, по аналогии с предыдущим этапом, будем обозначать через $k(z_1, \ldots, z_m)$. Предполагается, что промежуток $k(z_1, \ldots, z_m)$ содержит свой левый конец; он содержит также свой правый конец, если и только если $z_1 = \ldots = z_m = 2^n - 1$. Длина $k(z_1, \ldots, z_m)$ равна, очевидно, 2^{-mn}.

Определим отображение φ следующим условием:

$$u = \varphi(x) \Leftrightarrow [x \in k(z_1, \ldots, z_m) \Rightarrow u \in K(z_1, \ldots, z_m), \ m \geqslant 1],$$

т. е. точка $\varphi(x)$ принадлежит всем кубам $K(z_1, \ldots, z_m)$ таким, что $x \in k(z_1, \ldots, z_m)$. Существование и единственность точки $\varphi(x)$ следует из известного принципа вложенных шаров [127]. В [238] указан способ нумерации кубов каждого разбиения числами $z_1, z_2, \ldots$, обеспечивающий непрерывность отображения φ. Будем считать, что в проведенном выше построении использован именно этот способ. Полученное отображение φ и есть отображение Пеано, определяющее известную кривую Пеано.

Переходим к построению отображения φ_m. Ясно, что 2^{mn} центров кубов m-го разбиения образуют равномерную ортогональную сетку в K, причем шаг сетки по любой координате равен 2^{-m}. Занумеруем узлы этой сетки, а вместе с ними кубы m-го разбиения с помощью одного индекса i. Для этого занумеруем сначала слева направо номерами $i = 0, 1, \ldots, 2^{mn} - 1$ все промежутки, составляющие m-е разбиение отрезка [0, 1]. Далее, обозначим через K_i куб $K(z_1, \ldots, z_m)$ и через v_i — его центр, если промежуток $k(z_1, \ldots, z_m)$ получил номер i. При этом, как показано в [238], кубы K_i и K_{i+1} всегда являются смежными, т. е. имеют общую грань.

Обозначим через w_i точку пересечения отрезка, соединяющего v_{i-1} и v_i с общей гранью кубов K_{i-1} и K_i, $i = 1, \ldots, 2^{mn} - 1$, через w_0 — проекцию центра v_0 на грань куба K_0, параллельную общей грани кубов K_0 и K_1, че-

рез $w_{2^{mn}}$ — проекцию центра $v_{2^{mn}-1}$ на грань куба $K_{2^{mn}-1}$, параллельную общей грани кубов $K_{2^{mn}-1}$ и $K_{2^{mn}-2}$. Рассмотрим отображение

$$\varphi_m(x) = \begin{cases} w_0 + (v_0 - w_0)\, x 2^{mn+1}, & 0 \leqslant x \leqslant \frac{1}{2^{mn+1}}; \\ v_{i-1} + (v_i - v_{i-1})\left(2^{mn}x - i + 1/2\right), & \frac{i-1/2}{2^{mn}} \leqslant \\ & \leqslant x \leqslant \frac{i+1/2}{2^{mn}}, \quad i = 1, \ldots, 2^{mn} - 1; \\ v_{2^{mn}-1} + \left(w_{2^{mn}} - v_{2^{mn}-1}\right)\left[2^{mn+1}(x-1) + 1\right], & \\ & 1 - \frac{1}{2^{mn+1}} \leqslant x \leqslant 1, \end{cases}$$

действующее из $[0, 1]$ в куб K. Образ отрезка $[(i - 1/2)2^{-mn}, (i + 1/2)2^{-mn}]$ при отображении φ_m является линейным отрезком, соединяющим точки v_{i-1} и v_i. Таким

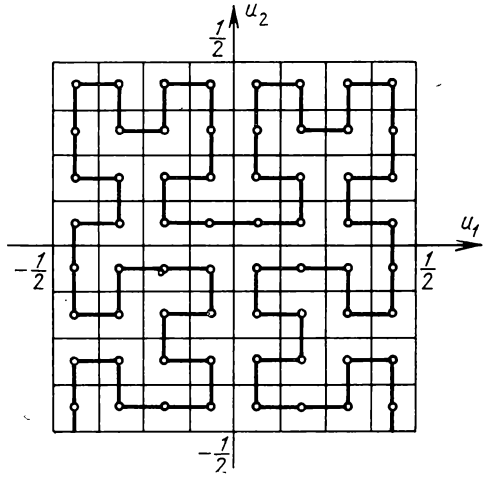

Рис. 6. Образ отрезка $[0, 1]$ при отображении φ_m

образом, $\{\varphi_m(x) \mid x \in [0, 1]\}$ есть кусочно линейная кривая, последовательно соединяющая центры смежных кубов m-го разбиения в порядке их нумерации. Нетрудно видеть, что $\varphi_m(x)$ отличается от $\varphi(x)$ по каждой координате не более чем на 2^{-m} при любом $x \in [0, 1]$. Рис. 6 соответствует случаю $n = 2$, $m = 3$.

2. Основной результат.

Теорема 6.1 [73]. *Пусть*

$$|f(x)-f(x')|\leqslant M\,\|x-x'\|,\quad x,x'\in K, \tag{6.1}$$

где $\|\cdot\|$ — *евклидова норма. Тогда*

$$\left|\int\limits_K f(u)\,du-\int\limits_0^1 f(\varphi_m(x))\,dx\right|\leqslant M\sqrt{n}\cdot 2^{-m}. \tag{6.2}$$

Доказательство. Обозначим через p_i прообраз точки v_i, а через q_j — прообраз точки w_j при отображении φ_m; $i=0,1,\ldots,2^{mn-1}$; $j=0,1,\ldots,2^{mn}$. Зафиксируем произвольное $i\in\{0,1,\ldots,2^{mn-1}\}$. Участок ломаной φ_m, лежащей в кубе K_i, определяется соотношением

$$\varphi_m(x)=\begin{cases}w_i+(v_i-w_i)(x-q_i)\,2^{mn+1}, & x\in[q_i,p_i],\\ v_i+(w_{i+1}-v_i)(x-p_i)\,2^{mn+1}, & x\in[p_i,q_{i+1}].\end{cases}$$

Сравним интегралы $\int\limits_{K_i} f(u)\,du$ и $\int\limits_{q_i}^{q_{i+1}} f(\varphi_m(x))\,dx$. В силу непрерывности f

$$\int\limits_{K_i} f(u)\,du=f(\xi_i)\,\mu(K_i)=f(\xi_i)\,2^{-mn},$$

где $\mu(K_i)$ — объем куба K_i, $\xi_i\in K_i$. В свою очередь,

$$\int\limits_{q_i}^{q_{i+1}} f(\varphi_m(x))\,dx=\int\limits_{q_i}^{p_i} f(\varphi_m(x))\,dx+\int\limits_{p_i}^{q_{i+1}} f(\varphi_m(x))\,dx=$$
$$=f(\eta_i)(p_i-q_i)+f(\zeta_i)(q_{i+1}-p_i)=2^{-mn-1}[f(\eta_i)+f(\zeta_i)],$$

где $\eta_i\in K_i$, $\zeta_i\in K_i$. Отсюда, учитывая (6.1) и тот факт, что диаметр куба K_i равен $2^{-m}\sqrt{N}$, получаем

$$\left|\int\limits_{K_i} f(u)\,du-\int\limits_{q_i}^{q_{i+1}} f(\varphi_m(x))\,dx\right|=$$
$$=2^{-mn-1}\,|2f(\xi_i)-f(\eta_i)-f(\zeta_i)|\leqslant 2^{-mn-1}\times$$
$$\times(|f(\xi_i)-f(\eta_i)|+|f(\xi_i)-f(\zeta_i)|)\leqslant 2^{-mn-m}M\sqrt{n}.$$

Остается просуммировать полученные неравенства при $i=0,1,\ldots,2^{mn-1}$. $\triangle$

Замечание. Если в (6.1) фигурирует произвольная (не обязательно евклидова) норма в n-мерном координатном пространстве, то теорема 6.1 останется справедливой, если правую часть в (6.2) заменить на Md_m, где d_m — диаметр кубов m-го разбиения (метрика, используемая при вычислении диаметра, определяется нормой, фигурирующей в (6.1)). △

3. Оптимальная квадратурная формула для класса функций, удовлетворяющих условию Гёльдера. Рассмотрим вопрос об интегрировании функции $g(x)=f(\varphi_m(x))$. В [238] показано, что функция g удовлетворяет на отрезке [0, 1] условию Гёльдера с показателем $1/n$:

$$|g(x)-g(x')|\leqslant M_0|x-x'|^{1/n},\quad x,\ x'\in[0,\ 1],\qquad (6.3)$$

где $M_0=4M\sqrt{n}$. Используем результат § 1 для построения оптимальной на классе (6.3) квадратурной формулы.

Теорема 6.2 [73]. *Оптимальная на классе* (6.3) *квадратурная формула с N узлами задается узлами $x_i^0=h(2^{-1/n}+i-1)$, $i=1,\ \ldots,\ N$, где $h=(N-1+2^{1-1/n})^{-1}$, и коэффициентами $p_1^0=p_N^0=(2^{-1/n}+1/2)h$, $p_2^0=\ldots=p_{N-1}^0=h$, а гарантированный ею результат равен*

$$\frac{M_0h^{1+1/n}\left(N-1+2^{1-1/n^2}\right)}{2^{1/n}(1+1/n)}.$$

Доказательство. В силу следствия из теоремы 1.2, наилучший гарантированный результат равен

$$M_0\min_{x_1,\ldots,x_N\in[0,1]}\int_0^1\min_{i=1,\ldots,N}|x-x_i|^{1/n}\,dx,\qquad (6.4)$$

а оптимальные узлы находятся из условия реализации внешнего минимума в (6.4). Положим $z_1=x_1$, $z_i=x_i-x_{i-1}$, $i=2,\ \ldots,\ N$, $z_{N+1}=1-x_N$. Тогда

$$\int_0^1\min_{i=1,\ldots,N}|x-x_i|^{1/n}\,dx=\int_{x_1-z_1}^{x_1+z_2/2}|x-x_1|^{1/n}\,dx+$$

$$+\sum_{i=2}^{N-1}\int_{x_i-z_i/2}^{x_i+z_{i+1}/2}|x-x_i|^{1/n}\,dx+\int_{x_N-z_N/2}^{x_N+z_{N+1}}|x-x_N|^{1/n}\,dx=$$

$$=\frac{2^{1/n}z_1^{1+1/n}+\sum\limits_{i=2}^{N}z_i^{1+1/n}+2^{1/n}z_{N+1}^{1+1/n}}{2^{1/n}(1+1/n)}.\qquad (6.5)$$

Минимизируя методом множителей Лагранжа (6.5) при условиях $\sum_{i=1}^{N+1} z_i = 1, z_i \geqslant 0,\ i = 1, \ldots, N+1$, находим оптимальные узлы и наилучший гарантированный результат. Оптимальные коэффициенты, в силу теоремы 1.2, выражаются формулами

$$p_1^0 = z_1^0 + \frac{z_2^0}{2},\quad p_i^0 = \frac{z_i^0 + z_{i+1}^0}{2},\quad i = 2, \ldots, N,\quad p_N^0 = \frac{z_N^0}{2} + z_{N+1}^0,$$

где $z_1^0, \ldots, z_{N+1}^0$ реализуют условный минимум в (6.5). △

4. Замечание о программной реализации алгоритмов вычисления кратных интегралов. Можно предложить ряд алгоритмов интегрирования функций многих переменных, основанных на замене вычисления $\int\limits_K f(u)\,du$ вычислением интеграла $\int\limits_0^1 f(\varphi_m(x))\,dx$ и различающихся способами интегрирования функций одной переменной. С точки зрения простоты программной реализации среди них следует выделить алгоритм, в котором интеграл $\int\limits_0^1 f(\varphi_m(x))\,dx$ находится с помощью построенной в п. 3 оптимальной квадратурной формулы.

Теорема 6.1 дает в качестве оценки неустранимой погрешности любого из указанных алгоритмов величину $M\sqrt{n}\,2^{-m}$. Эксперименты, проведенные на ЭВМ БЭСМ-6, показали, что время, затрачиваемое на вычисление $\varphi_m(x)$ при всевозможных значениях m и n, лежащих в пределах $1 \leqslant n \leqslant 10$, $4 \leqslant m \leqslant 20$, составляет сотые доли секунд. Однако выбор очень большого m с целью уменьшить неустранимую погрешность может привести к тому, что число 2^{mn} выйдет из диапазона представимых в ЭВМ чисел. При выборе m в каждом конкретном случае наряду с особенностями задачи следует учитывать и это обстоятельство.

ГЛАВА 3

ВОССТАНОВЛЕНИЕ ФУНКЦИЙ ПО ИХ ЗНАЧЕНИЯМ

В данной главе изучается задача восстановления функции по ее значениям в конечном числе точек. Построены детерминированные оптимальные пассивные алгоритмы. Показано, что применение последовательных алгоритмов не позволяет улучшить результат, гарантированный в классе всех пассивных алгоритмов. Для случая функций одной переменной построен последовательно-оптимальный алгоритм.

§ 1. Оптимальные пассивные алгоритмы

Задачи оптимального восстановления функции по информации о ней в конечном числе точек рассматриваются во многих работах. В основном, изучаются различные классы дифференцируемых [29, 34, 114, 132, 276, 321, 322, 344—346, 392, 393] или аналитических [193, 194, 276, 353] функций. Рассматриваются случаи как точной, так и приближенной [160, 161, 389] информации о функции. Вычислительные аспекты проблемы рассмотрены, например, в [334].

В большинстве из перечисленных работ предполагается, что точки, в которых вычисляются значения функции (или иная информация), фиксированы, т. е. необходимо выбрать лишь итоговую операцию (иными словами, необходимо проинтерполировать функцию по заданной информации). Мы рассмотрим такие постановки, в которых основное внимание уделяется вопросу оптимального выбора точек информационных вычислений.

1. Постановка задачи. Задача восстановления функции по ее значениям естественно возникает в следующей ситуации.

Пусть необходимо многократно (хотя бы приближенно) вычислять значения функции $f \in F$ в различных точках множества K, заранее не известно в каких. Ресурсы вычислителя ограничены, вследствие чего число N вычислений значений функции (информационных вычис-

лений) фиксировано. Вычислитель выбирает $x^N=(x_1, \ldots, x_N)$, вычисляет $y^N=(y_1, \ldots, y_N)$, где $y_j=f(x_j)$, $j=1, \ldots, N$, и, имея информацию $z^N=(x^N, y^N)$, выбирает функцию β: $K\to\mathbf{R}$. В дальнейшем вычисление значений функции f заменяется вычислением значений функции β. Говорят, что функция β восстанавливает функцию f по информации z^N. При этом естественно считать погрешностью величину $\sup\limits_{x\in K}|f(x)-\beta(x)|$.

Описанная ситуация полностью соответствует общей модели вычислений гл. 1, если положить

$$S(f)=f, \quad B=\{\beta|\beta\colon K\to\mathbf{R}\},$$
$$\varepsilon(\alpha, f)=\gamma(S(f), \beta)=\sup_{x\in K}|f(x)-\beta(x)|, \tag{1.1}$$

где $\alpha=(\tilde{x}_1, \ldots, \tilde{x}_N, \tilde{\beta})$, а функция $\beta=\tilde{\beta}(z^N)\in B$ находится по формулам (3.2) гл. 1. При этом $x_j(f)=f(x_j)$, $X_j=K$, $j=1, \ldots, N$, и произвольные итоговые операции считаются допустимыми.

Согласно теореме 4.3 гл. 1, задача построения оптимального по точности пассивного алгоритма сводится к отысканию величины

$$v_N\overset{\text{def}}{=}\inf_{x^N\in X^N}\ \sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}}\ \inf_{\beta\in B}\ \sup_{f\in F(z^N)}\ \sup_{x\in K}|f(x)-\beta(x)|, \tag{1.2}$$

а также вектора x_0^N, реализующего внешнюю нижнюю грань, и центральной итоговой операции $\tilde{\beta}_*$ (при фиксированном z^N элемент $\tilde{\beta}_*(z^N)$ реализует внутреннюю нижнюю грань в (1.2)).

2. Сведéние задачи построения оптимального пассивного алгоритма к задаче оптимального покрытия множества K. Далее в этом параграфе будем предполагать, что $F=F_\rho$, где F_ρ — функциональный класс, заданный некоторой квазиметрикой ρ. Используемые ниже миноранта φ_{1N} и мажоранта φ_{2N} для функций из $F(z^N)$ определены формулами (1.3) гл. 2 при $i=N$ (см. также леммы 1.1 и 1.2 гл. 2), а понятия радиуса $R(x^N)$ покрытия, заданного центрами $x_1, \ldots, x_N$, и радиуса R_N оптимального покрытия определены формулами (2.2), (2.3) гл. 2. Положим еще

$$\varphi_N(x)=\frac{1}{2}(\varphi_{1N}(x)+\varphi_{2N}(x)), \quad x\in K. \tag{1.3}$$

График функции φ_N в одномерном случае при $\rho(u, v) = M|u-v|$ изображен на рис. 3 (с. 102).

Теорема 1.1 [246]. *Центральная итоговая операция в задаче восстановления функции из класса F_ρ по ее значениям задается формулой*

$$\tilde{\beta}_*(z^N) = \varphi_N.$$

Результат, гарантированный пассивным алгоритмом $(x^N, \tilde{\beta}_*)$, равен радиусу $R(x^N)$ покрытия, заданного центрами $x_1, \ldots, x_N$. Наилучший гарантированный результат v_N равен радиусу оптимального покрытия R_N. Алгоритм $(x_0^N, \tilde{\beta}_*)$ оптимален по точности в множестве всех пассивных алгоритмов тогда и только тогда, когда вектор x_0^N задает оптимальное покрытие множества K.

Доказательство. Для отыскания центральной итоговой операции воспользуемся тем, что

$$\inf_{\beta\in B} \sup_{f\in F(z^N)} \sup_{x\in K} |f(x) - \beta(x)| =$$
$$= \inf_{\beta\in B} \sup_{x\in K} \sup_{f\in F(z^N)} |f(x) - \beta(x)| =$$
$$= \sup_{x\in K} \inf_{r\in \mathbf{R}} \sup_{f\in F(z^N)} |f(x) - r| =$$
$$= \sup_{x\in K} \inf_{r\in \mathbf{R}} \max\{\varphi_{2N}(x) - r, r - \varphi_{1N}(x)\} =$$
$$= \sup_{x\in K} \frac{1}{2}[\varphi_{2N}(x) - \varphi_{1N}(x)] \quad (1.4)$$

(первое и четвертое равенства очевидны, второе обосновывается леммой 4.1 гл. 1, третье — леммами 1.1 и 1.2 гл. 2), причем нижняя грань по r достигается для произвольного $x \in K$ при $r = \varphi_N(x)$. Поэтому, в силу второго утверждения леммы 4.1 гл. 1 и определения (см. (4.4) гл. 1), центральная итоговая операция задается формулой $\tilde{\beta}_*(z^N) = \varphi_N$.

Убедимся теперь, что

$$\sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}} \frac{1}{2}[\varphi_{2N}(x) - \varphi_{1N}(x)] = \min_{i=1,\ldots,N} \rho(x_i, x). \quad (1.5)$$

Действительно,

$$\sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}} \frac{1}{2}[\varphi_{2N}(x) - \varphi_{1N}(x)] =$$
$$= \frac{1}{2} \sup_{y^N\in\{y^N|F(z^N)\neq\varnothing\}} \Big[\min_{j=1,\ldots,N} (y_j + \rho(x, x_j)) -$$

$$- \max_{j=1,\ldots,N} (y_j - \rho(x, x_j))\Big] \leqslant$$
$$\leqslant \frac{1}{2} \sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \min_{j=1,\ldots,N} [(y_j + \rho(x, x_j)) -$$
$$- (y_j - \rho(x, x_j))] = \min_{j=1,\ldots,N} \rho(x, x_j).$$

С другой стороны, учитывая, что $(0, \ldots, 0) \in \{y^N | F(z^N) \neq \neq \varnothing\}$, имеем

$$\sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \frac{1}{2} [\varphi_{2N}(x) - \varphi_{1N}(x)] \geqslant$$
$$\geqslant \frac{1}{2} \Big[\min_{j=1,\ldots,N} (0 + \rho(x, x_j)) - \max_{j=1,\ldots,N} (0 - \rho(x, x_j))\Big] =$$
$$= \min_{j=1,\ldots,N} \rho(x, x_j).$$

Тем самым (1.5) доказано.

Преобразуем выражение для результата, гарантированного алгоритмом $(x_0^N, \widetilde{\beta}_*)$. В силу (1.4), (1.5) имеем

$$\sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \inf_{\beta \in B} \sup_{f \in F(z^N)} \sup_{x \in K} |f(x) - \beta(x)| =$$
$$= \sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \sup_{x \in K} \frac{1}{2} [\varphi_{2N}(x) - \varphi_{1N}(x)] =$$
$$= \sup_{x \in K} \sup_{y^N \in \{y^N | F(z^N) \neq \varnothing\}} \frac{1}{2} [\varphi_{2N}(x) - \varphi_{1N}(x)] =$$
$$= \sup_{x \in K} \min_{i=1,\ldots,N} \rho(x_i, x) = R(x^N).$$

Остальные утверждения теоремы теперь очевидны. △

3. О задаче оптимального покрытия. Рассмотрим некоторые вопросы, связанные с оптимальным покрытием множества K заданным числом ρ-шаров одинакового радиуса. Начнем со следующего замечания.

Если K — компакт, а функция ρ непрерывна, то оптимальное покрытие множества K заданным числом N ρ-шаров одинакового радиуса существует.

Справедливость этого утверждения следует из того, что функции $\min_{i=1,\ldots,N} \rho(x_i, x)$ и $R(x^N) = \max_{x \in K} \min_{i=1,\ldots,N} \rho(x_i, x)$ непрерывны соответственно по x и по x^N, следовательно, минимум функции $R(x^N)$ достигается при некотором x_0^N.

Приведем для конкретных ρ и K решение задачи построения оптимального покрытия.

Пусть $u=(u^1, \ldots, u^n)$, $M>0$, $k_1 \geqslant 0, \ldots, k_n \geqslant 0$,

$$K = \{u \mid a^i \leqslant u^i \leqslant b^i,\ i=1, \ldots, n\},$$

$$\rho(u, v) = M \max_{i=1,\ldots,n} k_i |u^i - v^i|. \tag{1.6}$$

Тогда ρ-шаром $\{u|\rho(z, u) \leqslant r\}$ радиуса r с центром в z является, очевидно, координатный параллелепипед

$$\left\{u \mid z^i - \frac{r}{Mk_i} \leqslant u^i \leqslant z^i + \frac{r}{Mk_i},\ i=1, \ldots, n\right\}.$$

Положим $c^i = b^i - a^i$, $d^i = 2/(Mk_i)$, $i=1, \ldots, n$.

Задача заключается в построении покрытия координатного параллелепипеда с ребрами $c^1, \ldots, c^n$ такими N координатными параллелепипедами с ребрами $rd^1, \ldots, rd^n$, что r минимально. Ясно, что минимальное значение r равно при этом радиусу оптимального покрытия R_N. Поэтому

$$R_N = \min\left\{r \left| \prod_{i=1}^{n} \left\lceil \frac{c^i}{rd^i} \right\rceil \leqslant N\right.\right\}. \tag{1.7}$$

Дадим теперь основанный на соотношении (1.7) алгоритм решения задачи (1.6). Пусть $\bar{r} = \left(\frac{1}{N}\prod_{i=1}^{n} \frac{c^i}{d^i}\right)^n$. Очевидно, что $R_N \geqslant \bar{r}$. Кроме того, из (1.7) следует, что хотя бы одно из чисел $c^i/(R_N d^i)$ должно быть целым, $i=1, \ldots, n$. Положив $m_i = \lceil c^i/(\bar{r}d^i)\rceil$, занумеруем в порядке возрастания корни $m_1 + \ldots + m_n$ уравнений

$$\frac{c^1}{rd^1} = m_1, \quad \frac{c^1}{rd^1} = m_1 - 1, \quad \ldots, \quad \frac{c^1}{rd^1} = 1,$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$\frac{c^n}{rd^n} = m_n, \quad \frac{c^n}{rd^n} = m_n - 1, \quad \ldots, \quad \frac{c^n}{rd^n} = 1.$$

Получим $m_1 + \ldots + m_n$ чисел $r_1 \leqslant r_2 \leqslant \ldots \leqslant r_{m_1+\ldots+m_n}$. Будем в порядке возрастания индекса j проверять выполнение условия $\prod_{i=1}^{n} \lceil c^i/(r_j d^i)\rceil \leqslant N$. Если это условие впервые окажется выполненным при $j=j_0$, то $R_N = r_{j_0}$. Итак, решение задачи в случае (1.6) найдено.

Если K — координатный куб и в (1.6) $k_1 = \ldots = k_n = 1$, то решение задачи оптимального покрытия легко находится в явном виде.

Теорема 1.2. *Пусть*

$$K = \{u \mid 0 \leqslant u^i \leqslant 1,\ i = 1, \ldots, n\},$$
$$\rho(u, v) = M \max_{i=1,\ldots,n} |u^i - v^i|.$$

Тогда $R_N = M/(2m)$, *где* $m = \left[\sqrt[n]{N}\right]$, m^n *из центров* $x_1^0, \ldots, x_N^0$ *оптимального покрытия совпадают с точками* $(j_1/(2m), \ldots, j_n/(2m))$, $j_1, \ldots, j_n \in \{1, 3, \ldots, 2m-1\}$, *остальные* $N - m^n$ — *произвольные точки.*

Доказательство. Предположим сначала, что $N = m^n$, m — целое. Тогда, проводя гиперплоскости $u^i = j/m$, $i = 1, \ldots, n$, $j = 1, \ldots, m-1$, получим разбиение K на m^n координатных кубов с длиной ребер $1/m$. Обозначим их центры через $x_1^0, \ldots, x_N^0$. Очевидно, что вектор $x_0^N = (x_1^0, \ldots, x_N^0)$ задает оптимальное покрытие, причем $R_N = M/(2m) = M\left(2\sqrt[n]{N}\right)$.

Пусть теперь $m^n < N < (m+1)^n$, m — целое. Покажем, что и в этом случае минимальная длина ребра кубов покрытия равна $1/m$. Для этого достаточно доказать следующее: если K покрыт координатными кубами с ребрами длины $l < 1/m$, то таких кубов не менее $(m+1)^n$.

Докажем это утверждение индукцией по размерности пространства n. В одномерном случае оно очевидно. Предположим, что это утверждение справедливо в пространстве $\mathbf{R}^{n-1}$, и докажем его справедливость для $\mathbf{R}^n$. Проведем гиперплоскости

$$u^1 = l + \varepsilon, \quad u^1 = 2(l + \varepsilon), \ldots, u_1 = m(l + \varepsilon),$$

где $\varepsilon > 0$ таково, что $l + \varepsilon < 1/m$. Сечения куба K этими гиперплоскостями обозначим через $T_1, T_2, \ldots, T_m$. Грань с несущей гиперплоскостью $u^1 = 0$ обозначим через T_0.

Пусть K покрыт координатными кубами с ребрами длины $l < 1/m$. Тогда грань T_0 покрыта $(n-1)$-мерными кубами (сечениями или основаниями соответствующих n-мерных кубов) с ребрами длины $l < 1/m$. Число таких $(n-1)$-мерных кубов по индукционному предположению не меньше $(m+1)^{n-1}$, причем ни один из соответствую-

щих n-мерных кубов, очевидно, не пересекается с сечением T_1. Повторяя рассуждения о покрытии T_0 для сечений $T_1, \ldots, T_m$, получим, что куб K покрыт не менее чем $(m+1)(m+1)^{n-1}=(m+1)^n$ координатными кубами с ребрами длины l. △

Следствие. *Если* $n=1$, $K=[0, 1]$, $\rho(u, v)=M|u-v|$, *то в задаче восстановления функции из класса* F_ρ *по ее значениям* $v_N=M/(2N)$, *а алгоритм* $(1/(2N), 3/(2N), \ldots, (2N-1)/(2N))$ *оптимален в множестве всех пассивных алгоритмов.* △

Если $\rho(u, v)=\|u-v\|$, K — ограниченное множество с границей нулевой меры и известно наименее плотное покрытие пространства $\mathbf{R}^n$ транслятами шара $\Omega(1)=\{x \mid \|x\| \leqslant 1\}$, то, применяя конструкцию, описанную в теореме 2.2 гл. 2, легко построить асимптотически оптимальную последовательность алгоритмов.

Оптимальные покрытия можно строить также, решая задачу

$$R(x^N)=\sup_{a\in K}\min_{i=1,\ldots,N}\rho(x_i, a)\to\min_{x_1,\ldots,x_N\in K}$$

численно. Так, в [35] удалось построить оптимальное покрытие квадрата K в случае, когда ρ — евклидова метрика, $N\leqslant 15$.

4. Совпадение наилучших гарантированных результатов в классах пассивных и последовательных алгоритмов. В этом пункте мы покажем, что для задачи восстановления функций из F_ρ по их значениям наилучший гарантированный в классе пассивных алгоритмов результат v_N совпадает с наилучшим гарантированным в классе всех последовательных алгоритмов результатом ε_N. Ряд утверждений такого рода получен в § 5 гл. 1, где предполагается, что S — функционал. Эти утверждения можно перенести на рассматриваемый случай $S(f)=f$ и тем самым установить совпадение v_N и ε_N. Мы установим этот факт независимо с помощью рассуждений несколько иного рода.

Считая, что в качестве итоговой операции применяется центральная итоговая операция $\widetilde{\beta}_*(z^N)=\varphi_N$ (см. (1.3) и теорему 1.1), будем рассматривать классы X^N (см. (3.8) гл. 1) — пассивных и $\widetilde{X}^N$ (см. (3.10) гл. 1) — всех последовательных алгоритмов. В нашем случае $X^N=K^N$. Критерием оценки эффективности алгоритмов будет

служить критерий (4.14) гл. 1, т. е.

$$\varepsilon(x^N, f) = \sup_{x \in K} |f(x) - \varphi_N(x)|, \quad \varepsilon(\tilde{x}^N, f) = \varepsilon(x^N, f), \quad (1.8)$$

где в первом равенстве $\varphi_N = (\varphi_{1N} + \varphi_{2N})/2$, $y_i = f(x_i)$, $i = 1, \ldots, N$,

$$\varphi_{1N}(x) = \max_{i=1,\ldots,N} \{y_i - \rho(x, x_i)\},$$

$$\varphi_{2N}(x) = \min_{i=1,\ldots,N} \{y_i + \rho(x, x_i)\},$$

а во втором равенстве вектор x^N определяется алгоритмом $\tilde{x}^N$ и функцией f по формулам (3.2) гл. 1.

Теорема 1.3. *Для задачи восстановления функций из F_ρ по их значениям наилучшие гарантированные результаты в классах X^N и $\tilde{X}^N$ совпадают, т. е.*

$$\inf_{\tilde{x}^N \in \tilde{X}^N} \sup_{f \in F} \varepsilon(\tilde{x}^N, f) = \inf_{x^N \in X^N} \sup_{f \in F} \varepsilon(x^N, f), \quad (1.9)$$

и любой оптимальный в X^N алгоритм оптимален, таким образом, и в $\tilde{X}^N$.

Доказательство. Будем обозначать левую часть (1.9) через ε_N, правую — через v_N. Так как $X^N \subset \tilde{X}^N$, то $\varepsilon_N \leqslant v_N$. Для завершения доказательства достаточно показать, что $\varepsilon_N \geqslant v_N$.

Обозначим δ-оптимальный в $\tilde{X}^N$ алгоритм через $\tilde{x}_\delta^N$, $\delta > 0$. Тогда

$$\sup_{f \in F_\rho} \varepsilon(\tilde{x}_\delta^N, f) < \varepsilon_N + \delta.$$

Пусть $\tilde{x}_\delta^N(\mathbf{0}) = x_\delta^N = (x_{1\delta}, \ldots, x_{N\delta})$, где $\mathbf{0}$ — функция, тождественно равная нулю на K, а через $\tilde{x}_\delta^N(\mathbf{0})$ обозначен вектор, полученный по формулам (3.2) гл. 1 при $\tilde{x}^N = \tilde{x}_\delta^N$, $f = \mathbf{0}$. В силу теоремы 1.1

$$v_N = R_N = \inf_{x^N \in K^N} \sup_{x \in K} \min_{i=1,\ldots,N} \rho(x_i, x),$$

и, следовательно,

$$\sup_{x \in K} \min_{i=1,\ldots,N} \rho(x_{i\delta}, x) \geqslant v_N.$$

Поэтому существует такая точка $a_\delta \in K$, что

$$\min_{i=1,\ldots,N} \rho(x_{i\delta}, a_\delta) \geqslant v_N - \delta. \quad (1.10)$$

Определим функцию

$$g_\delta(x)=\begin{cases} v_N-\delta-\rho(x, a_\delta) & \text{при} \quad \rho(x, a_\delta)\leqslant v_N-\delta, \\ 0 & \text{при} \quad \rho(x, a_\delta)\geqslant v_N-\delta. \end{cases}$$

Очевидно, что $g_\delta(x)=\max\{v_N-\delta-\rho(x, a_\delta), 0\}$. На основании лемм 2.1 и 2.7 гл. 1 заключаем, что $g_\delta\in F_\rho$. Из (1.10) следует, что $g_\delta(x_{i\delta})=0$, $i=1, \ldots, N$, и поэтому $\tilde{x}_\delta^N(g_\delta)=\tilde{x}_\delta^N(0)=x_\delta^N$, где через $\tilde{x}_\delta^N(g_\delta)$ обозначен вектор, полученный по формулам (3.2) гл. 1 при $\tilde{x}^N=\tilde{x}_\delta^N$, $f=g_\delta$. Наконец, в силу (1.8),

$$\varepsilon_N>\sup_{f\in F_\rho}\varepsilon(\tilde{x}_\delta^N, f)-\delta\geqslant\varepsilon(\tilde{x}_\delta^N, g_\delta)-\delta=\varepsilon(x_\delta^N, g_\delta)-\delta=$$
$$=\sup_{x\in K}|g_\delta(x)-\varphi_N(x)|-\delta=\sup_{x\in K}g_\delta(x)-\delta=v_N-2\delta,$$

где $\varphi_N(x)\equiv 0$, так как $y_i=g_\delta(x_{i\delta})=0$, $i=1, \ldots, N$. Переходя к пределу при $\delta\to 0$, получаем, что $\varepsilon_N\geqslant v_N$. $\triangle$

§ 2. Последовательно-оптимальные и оптимальные на один шаг алгоритмы восстановления функций

Пусть

$$F=\{f\mid |f(u)-f(v)|\leqslant M|u-v|, \quad u, v\in[a, b]\}. \quad (2.1)$$

По-прежнему будем предполагать, что информационные вычисления состоят в вычислении значений восстанавливаемой функции в точках множества $K=[a, b]$. Теорема 1.1 устанавливает в рассматриваемом случае $\rho(u, v)=M|u-v|$ вид центральной итоговой операции, а следствие из теоремы 1.2 — вид оптимального по точности пассивного алгоритма восстановления функций. Основная задача этого параграфа — построить последовательно-оптимальный по точности алгоритм для класса (2.1) функций одной переменной, удовлетворяющих условию Липшица. Решив ее, мы без труда построим оптимальный на один шаг алгоритм, а также алгоритм последовательно-оптимальный по числу информационных вычислений.

1. Оптимальные по точности алгоритмы для подклассов функционального класса (2.1). При построении последовательно-оптимальных по точности алгоритмов, как и в случае задачи численного интегрирования, необходимо найти оптимальные алгоритмы на подклассах класса (2.1) функций с фиксированными значениями на одном

или обоих концах отрезка $[a, b]$. Решение этой задачи дают леммы 2.1 и 2.2 (а также теорема 3.1), в формулировках которых число информационных вычислений предполагается равным n, $x_0 = a$, $x_{n+1} = b$, а через F обозначается функциональный класс (2.1). В этом случае $F(x_{n+1}, y_{n+1})$, $F(x_0, y_0)$, $F(x_0, x_{n+1}, y_0, y_{n+1})$ — подклассы класса F, содержащие функции с фиксированными значениями соответственно на правом, левом и обоих концах отрезка $[a, b]$.

В формулировках лемм не указываются итоговые операции. Во всех случаях в качестве итоговой операции используется центральная итоговая операция, вид которой легко установить на основе теоремы 1.1, учитывая имеющуюся априорную информацию о значениях функции на концах отрезка $[a, b]$. Именно,

$$\tilde{\beta}_*(x^n, x_{n+1}, y^n, y_{n+1}) = \\ = \frac{1}{2}\left(\max_{j=1,\ldots,n+1}\{y_j - M|x - x_j|\} + \min_{j=1,\ldots,n+1}\{y_j + M|x - x_j|\}\right),$$

$$\tilde{\beta}_*(x_0, x^n, y_0, y^n) = \\ = \frac{1}{2}\left(\max_{j=0,1,\ldots,n}\{y_j - M|x - x_j|\} + \min_{j=0,1,\ldots,n}\{y_j + M|x - x_j|\}\right),$$

$$\tilde{\beta}_*(x_0, x^n, x_{n+1}, y_0, y^n, y_{n+1}) = \\ = \frac{1}{2}\left(\max_{j=0,1,\ldots,n+1}\{y_j - M|x - x_j|\} + \right. \\ \left. + \min_{j=0,1,\ldots,n+1}\{y_j + M|x - x_j|\}\right)$$

— центральные итоговые операции соответственно для классов $F(x_{n+1}, y_{n+1})$, $F(x_0, y_0)$, $F(x_0, x_{n+1}, y_0, y_{n+1})$. Далее предполагается, что рассматриваются именно эти итоговые операции и множества алгоритмов X^N (см. (3.8) гл. 1) и $\tilde{X}^N$ (см. (3.10) гл. 1).

Лемма 2.1. *Алгоритм*

$$\left(a + \frac{b-a}{2n+1}, a + 3\frac{b-a}{2n+1}, \ldots, a + (2n-1)\frac{b-a}{2n+1}\right) \in X^n$$

оптимален по точности на классе $F(x_{n+1}, y_{n+1})$ в множествах всех пассивных алгоритмов X^n и всех последовательных алгоритмов $\tilde{X}^n$.

Алгоритм

$$\left(a + 2\frac{b-a}{2n+1}, a + 4\frac{b-a}{2n+1}, \ldots, a + 2n\frac{b-a}{2n+1}\right) \in X^n$$

оптимален по точности на классе $F(x_0, y_0)$ в множестве всех пассивных алгоритмов X^n и всех последовательных алгоритмов $\tilde{X}^n$.

Наилучший гарантированный результат в обоих случаях равен $M(b-a)/(2n+1)$. △

Доказательство леммы легко получается теми же рассуждениями, которые использованы при доказательстве теорем 1.1 и 1.3. При этом следует иметь в виду, что наихудшее для вычислителя «поведение природы» состоит в том, что $f(x_j)=y_{n+1}$ для класса $F(x_{n+1}, y_{n+1})$ и $f(x_j)=y_0$ для класса $F(x_0, y_0)$, где x_j — точки, в которых вычисляются значения функции, $j=1, \ldots, n$. Отметим еще, что класс $F(x_{n+1}, y_{n+1})$ симметричен относительно функции $f(x)\equiv y_{n+1}$, а класс $F(x_0, y_0)$ симметричен относительно функции $f(x)\equiv y_0$.

Итак, первое вычисление в соответствии с оптимальным алгоритмом проводится в любой из точек

$$\left\{a+r\frac{b-a}{2n+1}\,\middle|\,\begin{array}{l} r=1, 3, \ldots, 2n-1 \text{ для класса } F(b, y_b), \\ r=2, 4, \ldots, 2n \text{ для класса } F(a, y_a)\end{array}\right\}. \tag{2.2}$$

Гораздо сложнее обстоит дело для класса $F(x_0, x_{n+1}, y_0, y_{n+1})$, который при $y_0\neq y_{n+1}$ свойством центральной симметрии не обладает.

Используя ранее введенное обозначение $\varepsilon_N(z^i)$ (см. (6.7), (6.8) гл. 1) для наилучшего гарантированного результата, будем обозначать наилучший гарантированный на классе $F(x_0, x_{n+1}, y_0, y_{n+1})$ в множестве всех последовательных алгоритмов с n информационными вычислениями результат через $\varepsilon_{n+2}(x_0, x_{n+1}, y_0, y_{n+1})$. Достаточно рассмотреть случай $x_0=a=0$, $x_{n+1}=b=1$. Наилучший гарантированный в ситуации 0, 1, y_0, y_{n+1} результат зависит, очевидно, лишь от величины $|y_{n+1}-y_0|$ или связанной с ней взаимно однозначным соответствием величины $l=-|y_{n+1}-y_0|/M+1$ (см. рис. 7). Значения константы Липшица M и число вычислений n считаются при этом фиксированными. Обозначим наилучший гарантированный результат через $(M/2)W_n(l)$. Иными словами, положим

$$W_n(l)=\frac{2}{M}\varepsilon_{n+2}(0, 1, y_0, y_{n+1}).$$

Случай, когда значения функции известны на концах отрезка $[a, b]$, заменой переменных $x'=(x-a)/(b-a)$,

$y' = y$ сводится к рассматриваемому случаю $K = [0, 1]$. Если функция $f(x)$ удовлетворяет на $[a, b]$ условию Липшица с константой M, то функция $f(a + x'(b - a))$

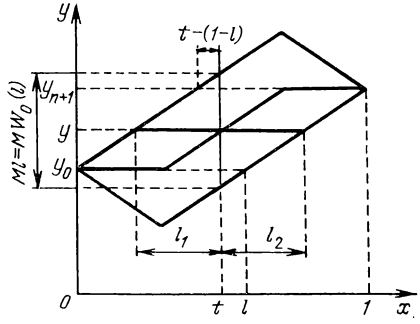

Рис. 7. К вопросу об отыскании наилучшего гарантированного результата в задаче восстановления функций с фиксированными значениями на концах отрезка [0, 1]

удовлетворяет, очевидно, по x' на [0, 1] условию Липшица с константой $M(b - a)$. Поэтому

$$\varepsilon_{n+2}(a, b, y_0, y_{n+1}) = \frac{M}{2}(b - a) W_n(l'), \tag{2.3}$$

где

$$l' = \frac{-|y_{n+1} - y_0|/M + b - a}{b - a}.$$

Лемма 2.2. *Наилучший гарантированный на классе $F(0, 1, y_0, y_{n+1})$ в множестве всех последовательных алгоритмов результат удовлетворяет рекуррентному уравнению*

$$W_n(l) = \min_{0<t<1} \max_{\substack{l_1, l_2 \\ 0 \leqslant l_1 \leqslant t, \\ 0 \leqslant l_2 \leqslant 1-t \\ l_1 + l_2 = l}} \min_{j=0,1,\ldots,n-1} \max \left\{ tW_j\left(\frac{l_1}{t}\right), \right.$$

$$\left. (1 - t) W_{n-1-j}\left(\frac{l_2}{1 - t}\right) \right\} \tag{2.4}$$

при

$$W_0(l) = l. \tag{2.5}$$

Первое вычисление в соответствии с оптимальным на классе $F(0, 1, y_0, y_{n+1})$ последовательным алгоритмом

следует проводить в точке, реализующей внешний минимум в правой части (2.4).

Доказательство. Пусть на первом шаге вычислитель выбирает точку $t \in [0, 1]$, причем

$$f(t) = y \in A(t) \overset{\text{def}}{=} [\varphi_{12}(t), \varphi_{22}(t)],$$

где $\varphi_{12}(t) = \max\{y_0 - M|x|, \quad y_{n+1} - M|x-1|\}$, $\varphi_{22}(t) = \min\{y_0 + M|x|, y_{n+1} + M|x-1|\}$ — соответственно миноранта и мажоранта, построенные по информации 0, 1, y_0, y_{n+1}. Далее воспользуемся соображениями, изложенными в п. 2 § 3 гл. 2. Именно, заметим, что для класса (2.1) значения, которые может принимать функция f в точках одного из интервалов $(0, t)$, $(t, 1)$, не изменяют множеств допустимых значений функции f в точках другого интервала (эти множества ограничены минорантой φ_{13} и мажорантой φ_{23}, построенными по информации 0, 1, t, y_0, y_{n+1}, y). Поэтому оптимальные действия вычислителя после первого шага состоят в том, чтобы оптимальным образом выбрать число $j \in \{0, 1, \ldots, n-1\}$ вычислений, которые предстоит провести на интервале $(0, t)$ (на интервале $(t, 1)$ будет проведено $n-1-j$ вычислений), и на последующих шагах оптимально выбирать точки для оставшихся $n-1$ вычислений.

Выражение для наилучшего гарантированного результата после первого шага можно записать в соответствии с (2.3), в виде

$$\varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y) = \\ = \frac{M}{2} \min_{j=0,1,\ldots,n-1} \max\left\{tW_j\left(\frac{-|y-y_0|/M+t}{t}\right), \right. \\ \left. (1-t)W_{n-1-j}\left(\frac{-|y-y_{n+1}|/M+1-t}{1-t}\right)\right\}. \qquad (2.6)$$

Поскольку оптимальное поведение в многошаговом процессе принятия решений складывается из оптимального решения на первом шаге и оптимального поведения в последующем, можно записать следующее уравнение для наилучшего гарантированного на классе $F(0, 1, y_0, y_{n+1})$ результата:

$$W_n(l) = \min_{0<t<1} \max_{y \in A(t)} \min_{j=0,1,\ldots,n-1} \max\left\{tW_j\left(\frac{-|y-y_0|/M+t}{t}\right), \right. \\ \left. (1-t)W_{n-1-j}\left(\frac{-|y-y_{n+1}|/M+1-t}{1-t}\right)\right\}. \qquad (2.7)$$

При этом, как видно из рис. 7,

$$W_0(l) = l. \tag{2.8}$$

Полученное рекуррентное уравнение (2.7) выражает известный принцип оптимальности динамического программирования [25, 26].

Для доказательства леммы теперь достаточно убедиться, что уравнения (2.4) и (2.7) эквивалентны. С этой целью покажем, что при произвольном n функция $W_n(l)$ не убывает по l.

При $n=0$ это следует из (2.8). Пусть $W_j(l)$ не убывает по l при $j=1, \ldots, n-1$. Зафиксируем $t \in (0, 1)$. Предположим для определенности, что $y_0 \leqslant y_{n+1}$ (в противном случае далее всюду отрезок $[y_0, y_{n+1}]$ надо заменить отрезком $[y_{n+1}, y_0]$). Используя (2.6) и предположение о неубывании функций W_j, $j=1, \ldots, n-1$, имеем

$$\begin{aligned} &\varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y) \leqslant \\ &\qquad \leqslant \varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y_0) \quad \text{при} \quad y \leqslant y_0, \\ &\varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y) \leqslant \\ &\qquad \leqslant \varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y_{n+1}) \quad \text{при} \quad y \geqslant y_{n+1}. \end{aligned} \tag{2.9}$$

Поэтому

$$W_n(l) = \frac{2}{M} \min_{0<t<1} \max_{y \in N(t)} \varepsilon_{n+2}(0, 1, t, y_0, y_{n+1}, y),$$

где $N(t) = A(t) \cap [y_0, y_{n+1}]$. Положим $l_1 = -|y-y_0|/M + t$, $l_2 = -|y-y_{n+1}|/M + 1 - t$. Тогда при $y \in N(t)$ имеем $l_1 + l_2 = l$. Если y пробегает множество $N(t)$, то l_1 при этом пробегает отрезок $[\max\{0, t-(1-l)\}, \min\{t, l\}]$ (см. рис. 7), а l_2 — отрезок $[\max\{0, l-t\}, \min\{1-t, l\}]$. Легко видеть, что ограничения на l_1, l_2 при изменении y в множестве $N(t)$ можно записать следующим образом: $0 \leqslant l_1 \leqslant t$, $0 \leqslant l_2 \leqslant 1-t$, $l_1 + l_2 = l$. Тем самым показана эквивалентность уравнений (2.4) и (2.7). Однако при доказательстве использованы неравенства (2.9), полученные в предположении о неубывании функции W_j.

Таким образом, для завершения доказательства всей леммы осталось завершить проводимое по индукции доказательство неубывания $W_n(l)$ по l. Но при любом фиксированном t функция, стоящая в правой части (2.4) под знаком внешнего минимума (а стало быть, и сама функция $W_n(l)$), не убывает по l. Последнее верно

в силу неубывания функций W_j, $j=0, 1, \ldots, n-1$, что имеет место согласно сделанному индукционному предположению. △

Основная трудность, которую приходится преодолеть при построении последовательно-оптимального алгоритма,— это необходимость найти в явном виде решение системы уравнений (2.4), (2.5). Последовательно-оптимальный алгоритм, при реализации которого приходилось бы на каждом шаге решать систему вида (2.4), (2.5) численно, обладал бы непомерно высокой комбинаторной сложностью. Решение системы (2.4), (2.5) в явном виде будет найдено в § 3. Согласно теореме 3.1,

$$W_n(l)=\begin{cases} l/k, & a_k(n)\leqslant l\leqslant b_k(n),\\ b_k(n)/k, & b_k(n)\leqslant l\leqslant a_{k+1}(n),\end{cases}\qquad k=1, \ldots, \lfloor n/2\rfloor+1,$$

где

$$a_1(n)=0,\ a_k(n)=\left(\frac{1}{2}\right)^{m_{k-1}(n)-1}\frac{k}{k+s_{k-1}(n)},$$
$$k=2, \ldots, \lfloor n/2\rfloor+1,$$
$$a_{\lfloor n/2\rfloor+2}=1;\ b_k(n)=\left(\frac{1}{2}\right)^{m_k(n)-1}\frac{k}{k+1+s_k(n)},$$
$$k=1, \ldots, \lfloor n/2\rfloor+1;$$

$m_k(n)$ — частное, $s_k(n)$ — остаток от деления n на k. При этом внешний минимум в (2.4) достигается при значении t, равном одному из значений

$$T_{n,r}(l)=\begin{cases}\dfrac{r}{k+s_{k-1}(n)} \text{ при } b_{k-1}(n)\leqslant l\leqslant b_k(n),\\ \qquad\qquad k=2, \ldots, \lfloor n/2\rfloor+2,\\ \text{любое число из } [0, 1] \text{ при } 0\leqslant l\leqslant b_1(n),\end{cases}\tag{2.10}$$

где $b_{\lfloor n/2\rfloor+2}(n)=1$, $r=1, \ldots, k+s_{k-1}(n)-1$.

2. Последовательно-оптимальный по точности алгоритм. Для упрощения обозначений будем считать, что $K=[0, 1]$. В силу определения последовательно-оптимального алгоритма, теоремы 1.3 и следствия из теоремы 1.2,

$$x_1\in\left\{\frac{1}{2N}, \frac{3}{2N}, \ldots, \frac{2N-1}{2N}\right\},\tag{2.11}$$

причем любую из точек (2.11) можно выбирать в качестве точки первого информационного вычисления.

Пусть i вычислений в точках $x_1, \ldots, x_i$ проведено, $x_{i1} < \ldots < x_{ii}$, где $x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$. Для построения последовательно-оптимального алгоритма, так же как и в задаче численного интегрирования, достаточно оптимальным образом распределить оставшиеся $N-i$ вычислений по промежуткам

$$[0, x_{i1}), (x_{i1}, x_{i2}), \ldots, (x_{i,i-1}, x_{ii}), (x_{ii}, 1], \quad (2.12)$$

найти оптимальные последовательные алгоритмы для функциональных классов, соответствующих этим промежуткам, и принять за x_{i+1} точку первого информационного вычисления любого из этих алгоритмов.

Сказанное, согласно леммам 2.1, 2.2, приводит к необходимости найти набор чисел $n_1^i, \ldots, n_{i+1}^i$ (j-е число в наборе — число информационных вычислений на j-м из промежутков (2.12)), реализующих минимум в задаче отыскания

$$\varepsilon_N(z^i) = \frac{M}{2} \min_{\substack{n_1,\ldots,n_{i+1}\in\{0,1,2,\ldots\} \\ \sum_{j=1}^{i+1} n_j = N-i}} \max\left\{\frac{2x_{i1}}{2n_1+1}, \right.$$

$$\left. (x_{i2}-x_{i1}) W_{n_2}(l_{i2}), \ldots, (x_{ii}-x_{i,i-1}) W_{n_i}(l_{ii}), \frac{2(1-x_{ii})}{2n_{i+1}+1}\right\}, \quad (2.13)$$

где

$$l_{ij} = \frac{-|f(x_{ij}) - f(x_{i,j-1})|/M + x_{ij} - x_{i,j-1}}{x_{ij} - x_{i,j-1}}.$$

После того как решение $n_1^i, \ldots, n_{i+1}^i$ найдено, за x_{i+1} можно принять любую точку из объединения наборов, определяемых соотношениями (2.2), (2.10), т. е.

$$x_{i+1} \in \left\{x_{i1} \frac{r_1}{2n_1^i+1} \middle| r_1 = 1, 3, \ldots, 2n_1^i - 1\right\} \cup$$

$$\cup \left(\bigcup_{j=2}^{i} \{x_{i,j-1} + (x_{ij} - x_{i,j-1}) T_{n_j^i, r_j}(l_{ij}) \mid r_j = \right.$$

$$\left. = 1, \ldots, k + s_{k-1}(n_j^i) - 1\} \right) \cup$$

$$\cup \left\{x_{ii} + (1-x_{ii}) \frac{r_{i+1}}{2n_{i+1}^i+1} \middle| r_{i+1} = 2, 4, \ldots, 2n_{i+1}^i\right\}, \quad (2.14)$$

где $i=1, \ldots, N-1$; число k, согласно формуле (2.10), определяется условием $b_{k-1}(n_j^i) \leqslant l_{ij} \leqslant b_k(n_j^i)$; наконец, предполагается, что если некоторые из чисел $n_1^i, \ldots, n_{i+1}^i$ равны нулю, то соответствующие наборы пусты.

Леммы 2.1, 2.2 и последующие построения дают полное доказательство следующей теоремы.

Теорема 2.1 [246]. *Алгоритм* (2.11), (2.14) *последовательно-оптимален по точности на классе* (2.1) *с* $[a, b]=[0, 1]$. △

3. Решение вспомогательных задач целочисленной оптимизации. Перейдем к вопросу о решении последовательности задач (2.13). Вводя при фиксированных x_{ij}, l_{ij} обозначения [1])

$$\alpha_1(n_1)=\frac{2x_{i1}}{2n_1+1}, \quad \alpha_j(n_j)=(x_{ij}-x_{i,j-1})\,W_{n_j}(l_{ij}),\ j=2, \ldots, i,$$

$$\alpha_{i+1}(n_{i+1})=\frac{2(1-x_{ii})}{2n_{i+1}+1}, \tag{2.15}$$

перепишем задачу (2.13) в виде

$$\max_{t=1,\ldots,i+1} \alpha_t(n_t) \to \min_{\substack{n_1,\ldots,n_{i+1}\in\{0,1,2,\ldots\} \\ \sum_{j=1}^{i+1} n_j=N-i}}. \tag{2.16}$$

Пусть $n_1^{i-1}, \ldots, n_i^{i-1}$ — решение задачи отыскания $\varepsilon_N(z^{i-1})$, возникающей при построении точки x_i, и для определенности $x_i \in (x_{i-1,j-1}, x_{i-1,j})$. Тогда $x_{i-1,1}=x_{i1}, \ldots$ $\ldots, x_{i-1,j-1}=x_{i,j-1}, x_i=x_{ij}, x_{i-1,j}=x_{i,j+1}, \ldots, x_{i-1,i-1}=x_{ii}$. В качестве начального приближения к решению $n_1^i, \ldots$ $\ldots, n_{i+1}^i$ выберем набор целых чисел $n_1^{i0}, \ldots, n_{i+1}^{i0}$, определяемый следующим образом:

$$n_1^{i0}=n_1^{i-1}, \ldots, n_{j-1}^{i0}=n_{j-1}^{i-1}, \ n_{j+2}^{i0}=n_{j+1}^{i-1}, \ldots, n_{i+1}^{i0}=n_i^{i-1}.$$

Числа n_j^{i0}, n_{j+1}^{i0} определяются как минимальные целые числа, для которых

$$\alpha_j(n_j^{i0}) \leqslant \frac{2}{M}\varepsilon_N(z^{i-1}), \quad \alpha_{j+1}(n_{j+1}^{i0}) \leqslant \frac{2}{M}\varepsilon_N(z^{i-1}).$$

---

[1]) Конечно, функции $\alpha_1, \ldots, \alpha_{i+1}$ зависят от номера шага, однако для упрощения обозначений мы дополнительный индекс не вводим.

Присвоим переменным $n_1, \ldots, n_{i+1}$ начальные значения $n_1^{i0}, \ldots, n_{i+1}^{i0} : n_1 := n_1^{i0}, \ldots, n_{i+1} := n_{i+1}^{i0}$. Положим $\Delta n^i = n_j^{i-1} - 1 - n_j^{i0} - n_{j+1}^{i0}$ и выполним Δn^i раз оператор $n_s := n_s + 1$, где $s \in \{1, \ldots, i+1\}$ — какой-либо (например наименьший) из индексов, для которых $\alpha_s(n_s) = \max\limits_{t=1,\ldots,i+1} \alpha_t(n_t)$.

Полученные значения $n_1, \ldots, n_{i+1}$ совпадают, как нетрудно видеть, с искомым решением $n_1^i, \ldots, n_{i+1}^i$, причем

$$\varepsilon_N(z^i) = \frac{M}{2} \max_{t=1,\ldots,i+1} \alpha_t(n_t^i). \tag{2.17}$$

Отметим, что если $i = N$, то из (2.17) немедленно получаем выражение для оценочной точности (апостериорного результата, гарантированного алгоритмом по окончании информационных вычислений)

$$\varepsilon_N(z^N) = \frac{M}{2} \max_{t=1,\ldots,N+1} \alpha_t(0). \tag{2.18}$$

4. Оптимальный на один шаг алгоритм. Для оптимального выбора точки x_1 достаточно решить задачу минимизации гарантированного после первого шага результата (2.18):

$$\varepsilon_1(z^1) = \max\{Mx_1,\ M(1 - x_1)\} \to \min, \quad x_1 \in [0,\ 1],$$

откуда

$$x_1 = 1/2. \tag{2.19}$$

Решая задачу (2.16) для $N - i = 1$, получаем из (2.14)

$$x_{i+1} = \begin{cases} x_{i1}/3, & \text{если} \quad \alpha_1(0) = m, \\ (x_{i,k-1} + x_{ik})/2, & \text{если} \quad \alpha_k(0) = m, \quad k \in \{2, \ldots, i\}, \\ x_{ii} + 2(1 - x_{ii})/3, & \text{если} \quad \alpha_{i+1}(0) = m, \end{cases} \tag{2.20}$$

где $m = \max\limits_{t=1,\ldots,i+1} \alpha_t(0)$, а величины $\alpha_t(0)$ определяются формулами (2.15). Тем самым доказана

Теорема 2.2. *Алгоритм* (2.19), (2.20) *оптимален на один шаг на классе* (2.1) *с* $[a, b] = [0, 1]$. △

5. Алгоритм, последовательно-оптимальный по числу информационных вычислений. Случай фиксированной точности ε проще случая фиксированного числа информационных вычислений N в том смысле, что при по-

строении последовательно-оптимального по числу информационных вычислений алгоритма отпадает необходимость в решении задач целочисленного программирования вида (2.16).

В силу теоремы 1.3 наилучший гарантированный в множестве всех последовательных алгоритмов результат ε_N совпадает с наилучшим гарантированным в множестве пассивных алгоритмов результатом v_N, причем, согласно следствию из теоремы 1.2, $v_N = M/(2N)$. Поэтому

$$N_\varepsilon = \min\{N \mid \varepsilon_N \leqslant \varepsilon\} = \min\left\{N \,\middle|\, \frac{M}{2N} \leqslant \varepsilon\right\} = \left\lceil \frac{M}{2\varepsilon} \right\rceil.$$

В соответствии с определениями оптимального по числу информационных вычислений алгоритма (см. п. 13 § 4 гл. 1) и последовательно-оптимального по числу информационных вычислений алгоритма (см. п. 4 § 6 гл. 1) выбор точки x_1 последовательно-оптимальным алгоритмом определяется формулой (2.11) при $N = N_\varepsilon$.

После i шагов проверяется выполнение критерия остановки (4.20) гл. 1. В рассматриваемой задаче выполнение критерия остановки в соответствии с (2.18) означает выполнение неравенства

$$\varepsilon_i(z^i) = \frac{M}{2} \max_{t=1,\ldots,i+1} \alpha_t(0) \leqslant \varepsilon.$$

Если критерий остановки не выполнен, то целые числа $n_1^i, \ldots, n_{i+1}^i$ определяются условием $M\alpha_t(n_t^i)/2 \leqslant \varepsilon$, причем либо $n_t^i = 0$, либо $M\alpha_t(n_t^i - 1)/2 > \varepsilon$, $t = 1, \ldots, i+1$, после чего точка x_{i+1} выбирается по формуле (2.14).

6. О схеме построения и о реализации последовательно-оптимальных алгоритмов. Заметим, что последовательно-оптимальные алгоритмы для задачи восстановления функций были построены по той же схеме, что и для задачи интегрирования. Узловые моменты этой схемы выделены в п. 7 § 3 гл. 2. Там же перечислены свойства моделей вычислений, позволяющие применять разработанную схему. Высказанные применительно к задаче интегрирования соображения применимы и к задаче восстановления функции по ее значениям. Однако в задаче восстановления построение оптимальных алгоритмов для функциональных классов, определяемых фиксированием информации на концах отрезка, может оказаться значи-

тельно более трудным, чем в задаче численного интегрирования.

При реализации последовательно-оптимальных алгоритмов следует разумно распорядиться произволом, допустимым в соответствии с формулами (2.11), (2.14) при выборе точки очередного информационного вычисления. Эту точку можно выбирать, например, случайно равновероятно из набора всех допустимых точек, можно выбирать точку из допустимого набора, ближайшую к точке, указываемой оптимальным на один шаг алгоритмом, и т. д.

Приведем результаты применения последовательно-оптимального по точности алгоритма в варианте, когда из допустимого набора (2.14) выбирается точка, ближайшая к точке, указываемой оптимальным на один шаг алгоритмом, к одной из случайных функций тестового семейства $f(x)=c_0+c_1x+c_2x^2/2+\ldots+c_nx^n/n!$, где $x\in\in[a,\ b]=[-2,\ 2]$, $c_0,\ c_1,\ \ldots,\ c_n$ — случайные числа, равномерно распределенные на отрезке $[-3,\ 3]$, $n=4$. При этом константа Липшица M вычислялась алгоритмом по формуле $M=\max\limits_{j=0,1,\ldots,160}|f'(-2+j/40)|$.

В первой строке табл. 4 указано число информационных вычислений N. Во второй строке указана точность

Таблица 4

N	10	30	50	80
v_N	0.6117	0.2039	0.1223	0.0765
$\varepsilon_N(z^N)$	0.5942	0.1594	0.0967	0.0543
$\varepsilon_N(z^N)/v_N$	0.9715	0.7820	0.7904	0.7106

$v_N=M(b-a)/(2N)$, гарантируемая оптимальным в множествах X^N и $\tilde{X}^N$ алгоритмом

$$\left(a+\frac{b-a}{2N},\ a+3\frac{b-a}{2N},\ \ldots,a+(2N-1)\frac{b-a}{2N}\right)$$

(эта точность совпадает с апостериорной точностью, гарантируемой по окончании информационных вычислений); в третьей строке — вычисляемая по формуле (2.18) апостериорная точность, гарантируемая по окончании информационных вычислений последовательно-оптимальным по точности алгоритмом; в четвертой — отношения

чисел из третьей строки к соответствующим числам из второй.

Приведенные результаты показывают, что, как и в задаче численного интегрирования, в задаче восстановления функции по ее значениям последовательно-оптимальный алгоритм обладает существенным преимуществом по сравнению с оптимальным алгоритмом.

§ 3. Решение многошаговой антагонистической игры, связанной с задачей оптимального восстановления

Применяя принцип оптимальности для многошагового процесса принятия решения, в § 2 задачу построения последовательно-оптимального алгоритма для задачи восстановления функции по ее значениям удалось свести к решению системы рекуррентных уравнений (лемма 2.2)

$$W_n(l) = \min_{0<t<1} \max_{\substack{l_1, l_2 \\ 0 \leqslant l_1 \leqslant t, 0 \leqslant l_2 \leqslant 1-t}} \min_{j=0,1,\ldots,n-1} \max \left\{ tW_j\left(\frac{l_1}{t}\right), \right.$$

$$\left. (1-t) W_{n-1-j}\left(\frac{l_2}{1-t}\right) \right\}, \qquad (3.1)$$

$$W_0(l) = l. \qquad (3.2)$$

В данном параграфе решение системы (3.1), (3.2) найдено в явном виде. Метод получения решения состоит в сведении задачи к многошаговой антагонистической игре с полной информацией и отыскании седловой точки игры.

Решение системы (3.1), (3.2) и соответствующей игры представляет также известный самостоятельный интерес, поскольку достаточно развитая теория систем рекуррентных уравнений разработана лишь в линейном случае (см., например, [196, 395]), примеры же решения в явном виде нелинейных систем достаточно редки [25, 26, 291], так же как и примеры решения в явном виде многошаговых игр с полной информацией и континуумами альтернатив у каждого из игроков [206].

1. Сведение задачи к антагонистической игре, формулировка и обсуждение основной теоремы. Из вида уравнений (3.1), (3.2) следует, что величина $W_n(l)$ является значением следующей многошаговой антагонистической игры, которую мы будем называть *игрой восстановления*.

На первом шаге минимизирующий игрок выбирает точку $x_1 \in (0, 1)$ (в обозначениях (3.1) — точку t), разбивая тем самым отрезок $[0, 1]$ на отрезки $[0, x_1]$, $[x_1, 1]$, а максимизирующий игрок, зная x_1, приписывает этим отрезкам неотрицательные числа l_1, l_2 соответственно, при этом число, приписываемое отрезку, не должно превосходить его длины и $l_1 + l_2 = l$, где $l \in [0, 1]$ — заранее фиксированное число, известное обоим игрокам.

На втором шаге минимизирующий игрок выбирает точку $x_2 \in (0, x_1) \cup (x_1, 1)$, разбивая тем самым один из отрезков первого шага на два, а максимизирующий игрок, зная x_1, x_2, приписывает этим двум отрезкам неотрицательные числа, каждое из которых не превосходит длины своего отрезка, а сумма равна числу, приписанному объединению этих отрезков на первом шаге, и т. д.

После того как все n шагов сделаны, максимизирующий игрок получает от минимизирующего максимальную из величин, приписанных к $n+1$ отрезкам, на которые разбит отрезок $[0, 1]$.

Решение сформулированной игры дает следующая теорема, являющаяся основным результатом § 2, 3.

Теорема 3.1 [245]. *Значение игры $W_n(l)$ зависит от n и l следующим образом:*

$$W_n(l) = \begin{cases} l/k, & a_k(n) \leqslant l \leqslant b_k(n), \\ b_k(n)/k, & b_k(n) \leqslant l \leqslant a_{k+1}(n), \quad k = 1, \ldots, [n/2]+1, \end{cases} \tag{3.3}$$

где

$$a_1(n) = 0, \quad a_k(n) = \left(\frac{1}{2}\right)^{m_{k-1}(n)-1} \frac{k}{k + s_{k-1}(n)},$$

$$k = 2, \ldots, \left[\frac{n}{2}\right] + 1,$$

$$a_{[n/2]} = 1; \quad b_k(n) = \left(\frac{1}{2}\right)^{m_k(n)-1} \frac{k}{k + 1 + s_k(n)},$$

$$k = 1, \ldots, \left[\frac{n}{2}\right] + 1;$$

$m_k(n)$ — частное, $s_k(n)$ — остаток от деления n на k.

Оптимальным выбором минимизирующего игрока на первом шаге и соответственно реализацией внешнего

минимума в (3.1) *является любая из точек*

$$T_{n,r}(l)=\begin{cases}\dfrac{r}{k+s_{k-1}(n)} \text{ при } b_{k-1}(n)\leqslant l\leqslant b_k(n),\ k=2,\ldots,\left[\dfrac{n}{2}\right]+2,\\ \text{любое число из } [0,1] \text{ при } 0\leqslant l\leqslant b_1(n),\end{cases} \tag{3.4}$$

где $b_{\lfloor n/2\rfloor+2}(n)=1$, $r=1,\ldots,k+s_{k-1}(n)-1$.

Оптимальным выбором максимизирующего игрока на первом шаге и соответственно реализацией максимума по l_1, l_2 *в* (3.1) *являются величины* $l_1(x_1)$, $l_2(x_1)$[1]), *где* $l_2(x_1)=l-l_1(x_1)$,

$$l_1(x_1)=$$

$$=\begin{cases}0 & \text{в случае } l\leqslant a_2(n),\ x_1\leqslant 1/2,\\ 0 & \text{в случае } (3.6),\ x_1\leqslant\dfrac{1}{k+1+s_k(n)},\\ (t-1)\dfrac{l}{k} & \text{в случае } (3.6),(3.7),\ s_k(n)\leqslant k-t+1,\\ \dfrac{t-1}{2}\cdot\dfrac{l}{k} & \text{в случае } (3.6),(3.7),\ t-1 \text{ четно},\ s_k(n)\geqslant\dfrac{t-1}{2},\\ \dfrac{t}{2}\cdot\dfrac{l}{k} & \text{в случае } (3.6),(3.7),\ t \text{ четно},\ s_k(n)\geqslant\dfrac{t}{2}-1,\\ l-l_1(1-x_1) & \text{в случае } x_1\geqslant 1/2,\end{cases} \tag{3.5}$$

при этом условия (3.6), (3.7) *состоят в следующем*:

$$a_k(n)\leqslant l\leqslant a_{k+1}(n),\qquad k=2,\ldots,\left[\frac{n}{2}\right]+1, \tag{3.6}$$

$$\frac{t-1}{k+1+s_k(n)}\leqslant x_1\leqslant\frac{t}{k+1+s_k(n)},\quad 1\leqslant t-1\leqslant\frac{k+1+s_k(n)}{2}. \tag{3.7}$$

Прежде чем доказывать сформулированную теорему, отметим следующее. Зная оптимальные выборы (3.4), (3.5) игроков на первом шаге, легко индуктивно построить оптимальные стратегии игроков в игре восстановления, которые также последовательно-оптимальны, т. е. не только гарантируют игрокам получение величины $W_n(l)$,

---

[1]) Напомним, что в целях сохранения единства обозначений при рассмотрении игры восстановления мы всюду заменяем обозначение t в (3.1) на x_1.

но и осуществляют оптимальное наказание противника за совершенные ошибки в случае, если противник применяет неоптимальную стратегию.

Для построения таких стратегий достаточно в ситуации после i шагов уметь оптимальным образом распределять число оставшихся $n-i$ выборов минимизирующего игрока по множествам (x_{i0}, x_{i1}), (x_{i1}, x_{i2}), ..., $(x_{i,i-1}, x_{ii})$, $(x_{ii}, x_{i,i+1})$, где $x_{i0}=0$, $x_{i,i+1}=1$; x_{i1}, ..., x_{ii} — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания, $x_1, \ldots, x_i$ — точки, выбранные минимизирующим игроком. Для распределения выборов минимизирующего игрока необходимо решить задачу отыскания набора целых неотрицательных чисел $n_1, \ldots, n_{i+1}$, реализующего

$$\min_{\substack{n_1,\ldots,n_{i+1}\\ n_1+\ldots+n_{i+1}=n-i}} \max_{j=1,\ldots,i+1} (x_{ij}-x_{i,j-1})\, W_{n_j}\left(\frac{l_{ij}}{x_{ij}-x_{i,j-1}}\right),$$

где l_{ij} — число, приписанное максимизирующим игроком отрезку $[x_{i,j-1}, x_{ij}]$. Алгоритм решения подобной задачи приведен в п. 3 § 2. Если $n_1^i, \ldots, n_{i+1}^i$ — решение задачи, то минимизирующему игроку на $(i+1)$-м шаге для того, чтобы остаться в рамках последовательно-оптимальных действий, достаточно в качестве x_{i+1} выбирать любую из точек $x_{i,j-1}+(x_{ij}-x_{i,j-1})\, T_{n_j^i, r_j}\left(\frac{l_{ij}}{x_{ij}-x_{i,j-1}}\right)$, где значения r_j определяются формулой (3.4), а $j=1, \ldots, i+1$ таково, что $n_j^i>0$; максимизирующему же игроку достаточно приписать отрезкам $[x_{i,j-1}, x_{i+1}]$, $[x_{i+1}, x_{ij}]$ соответственно числа

$$(x_{ij}-x_{i,j-1})\, l_1\left(\frac{x_{i+1}-x_{i,j-1}}{x_{ij}-x_{i,j-1}}\right),$$

$$l_{ij}-(x_{ij}-x_{i,j-1})\, l_1\left(\frac{x_{i+1}-x_{i,j-1}}{x_{ij}-x_{i,j-1}}\right),$$

где при использовании (3.5)—(3.7) следует положить $n=n_j^i$, $l=l_{ij}/(x_{ij}-x_{i,j-1})$.

Доказательство теоремы 3.1 содержится в пп. 2, 3. В п. 2 мы покажем, что минимизирующий игрок может, применяя на первом шаге указанную в теореме 3.1 стратегию, гарантировать себе при оптимальных действиях

в последующем проигрыш не более $W_n(l)$. В п. 3 мы убедимся, что максимизирующий игрок может, применяя на первом шаге указанную в теореме 3.1 стратегию, гарантировать себе при оптимальных действиях в последующем выигрыш не менее $W_n(l)$. Тем самым теорема будет доказана, поскольку в совокупности установленные факты означают, что указанные в теореме 3.1 выборы игроков на первом шаге оптимальны, а определенная формулой (3.3) величина $W_n(l)$ является значением игры.

2. Доказательство теоремы 3.1 — гарантированный результат минимизирующего игрока. В этом пункте докажем, что, выбирая на первом шаге в игре восстановления в качестве x_1 любую из точек (3.4), минимизирующий игрок обеспечивает себе при оптимальных действиях в последующем проигрыш не более (3.3).

Для $n=1$ доказываемое утверждение очевидно, так как

$$W_1(l)=\begin{cases} l, & l\leqslant 1/2,\\ 1/2, & l\geqslant 1/2,\end{cases}\quad T_{1,1}(l)=\begin{cases}\text{любое число из } [0,1], & l\leqslant 1/2,\\ 1/2, & l\geqslant 1/2.\end{cases}$$

Пусть для 2, ..., $(n-1)$-шаговых игр утверждение справедливо. Докажем его для n-шаговой игры.

При $l\leqslant b_1(n)$ доказывать нечего. Пусть

$$a_k(n)\leqslant l\leqslant b_k(n),\quad k=2,\ \ldots,\ [n/2]+1, \tag{3.8}$$

или

$$\left(\frac{1}{2}\right)^{m_{k-1}(n)-1}\frac{1}{k+s_{k-1}(n)}\leqslant\frac{l}{k}\leqslant\left(\frac{1}{2}\right)^{m_k(n)-1}\frac{1}{k+1+s_k(n)},$$
$$k=2,\ \ldots,\ [n/2]+1. \tag{3.9}$$

Достаточно убедиться, что минимизирующий игрок, выбирая на первом шаге $x_1=\dfrac{r}{k+s_{k-1}(n)}$, где r — произвольное из чисел 1, ..., $k+s_{k-1}(n)-1$, может при любом первом выборе l_1 максимизирующего игрока гарантировать себе проигрыш не более $W_n(l)=l/k$ (последнее равенство справедливо в силу (3.3), (3.8)).

Если $l_1\leqslant l/k$, то для обеспечения проигрыша не более l/k минимизирующему игроку следует все точки x_2, ..., x_n выбрать в отрезке $[x_1, 1]$, и потому остается доказать, учитывая предположение индукции, что в этом

случае

$$(1-x_1)\,W_{n-1}\left(\frac{l_2}{1-x_1}\right)\leqslant\frac{l}{k}.$$

Последнее неравенство будем доказывать в наихудшем для минизирующего игрока случае $l_2=l$, $r=1$; оно в этом случае принимает вид

$$\frac{k+s_{k-1}(n)-1}{k+s_{k-1}(n)}\,W_{n-1}\left(l\,\frac{k+s_{k-1}(n)}{k+s_{k-1}(n)-1}\right)\leqslant\frac{l}{k},$$

для чего необходимо и достаточно, чтобы

$$l\,\frac{k+s_{k-1}(n)}{k+s_{k-1}(n)-1}\geqslant a_k(n-1).$$

Последнее будет заведомо выполнено в силу (3.9), если

$$\left(\frac{1}{2}\right)^{m_{k-1}(n)-1}\frac{k}{k+s_{k-1}(n)}\cdot\frac{k+s_{k-1}(n)}{k+s_{k-1}(n)-1}\geqslant$$
$$\geqslant\left(\frac{1}{2}\right)^{m_{k-1}(n-1)-1}\frac{k}{k+s_{k-1}(n-1)},$$

или, что то же самое,

$$k+s_{k-1}(n-1)\geqslant 2^{m_{k-1}(n)-m_{k-1}(n-1)}(k+s_{k-1}(n)-1). \tag{3.10}$$

На самом деле левая и правая части (3.10) всегда связаны знаком равенства, ибо если $m_{k-1}(n)=m_{k-1}(n-1)+1$, то $s_{k-1}(n-1)=k-2$, $s_{k-1}(n)=0$, если же $m_{k-1}(n)=m_{k-1}(n-1)$, то $s_{k-1}(n)=s_{k-1}(n-1)+1$.

Обратимся к случаю

$$(t-1)\frac{l}{k}\leqslant l_1\leqslant t\,\frac{l}{k},\quad t=2,\ldots,k-1 \tag{3.11}$$

(случай $t=k$ в силу равноправия концов отрезка [0, 1] сводится к рассмотренному случаю $t=1$). Минимизирующему игроку при этом для того, чтобы не проиграть больше l/k при любом l_1, удовлетворяющем (3.11), достаточно из оставшихся $n-1$ выбрать n_1 точек в интервале $(0, x_1)$, где n_1 определяется в соответствии с индукционным предположением соотношениями

$$x_1W_{n_1}\left(t\,\frac{l}{k}\cdot\frac{1}{x_1}\right)\leqslant\frac{l}{k},\quad x_1W_{n_1-1}\left(t\,\frac{l}{k}\cdot\frac{1}{x_1}\right)>\frac{l}{k},$$

которые эквивалентны соответственно неравенствам

$$t\,\frac{l}{k}\cdot\frac{1}{x_1}\geqslant a_t(n_1), \tag{3.12}$$

$$t\,\frac{l}{k}\cdot\frac{1}{x_1}<a_t(n_1-1). \tag{3.13}$$

Поскольку $l-l_1\leqslant(k+1-t)\,\frac{l}{k}$, достаточно убедиться, что имеет место неравенство

$$(1-x_1)\,W_{n-1-n_1}\left(\frac{k+1-t}{1-x_1}\cdot\frac{l}{k}\right)\leqslant\frac{l}{k}$$

или эквивалентное неравенство

$$\frac{k+1-t}{1-x_1}\cdot\frac{l}{k}\geqslant a_{k+1-t}(n-1-n_1),$$

для чего в силу (3.9) достаточно проверить, что

$$\frac{k+1-t}{1-x_1}\left(\frac{1}{2}\right)^{m_{k-1}(n)-1}\frac{1}{k+s_{k-1}(n)}\geqslant a_{k+1-t}(n-1-n_1).$$

Подставляя сюда $x_1=\frac{r}{k+s_{k-1}(n)}$ и выражение для $a_{k+1-t}(n-1-n_1)$, получаем неравенство

$$k+s_{k-1}(n)-r\leqslant 2^{m_{k-t}(n-1-n_1)-m_{k-1}(n)}[k+1-t+ \\ +s_{k-1}(n-1-n_1)], \tag{3.14}$$

которое предстоит доказать.

Для доказательства воспользуемся следствием из неравенства (3.13), (3.9):

$$t\left(\frac{1}{2}\right)^{m_{k-1}(n)-1}\frac{1}{k+s_{k-1}(n)}\cdot\frac{k+s_{k-1}(n)}{r}< \\ <\left(\frac{1}{2}\right)^{m_{t-1}(n_1-1)-1}\frac{t}{t+s_{t-1}(n_1-1)},$$

которое представим в виде

$$2^{m_{t-1}(n_1-1)}[t+s_{t-1}(n_1-1)]<r2^{m_{k-1}(n)}. \tag{3.15}$$

Предположим, что

$$\frac{1}{2}<\frac{t-1}{r}<r. \tag{3.16}$$

Тогда $0\leqslant r-t\leqslant t-3$, и потому $n_1^0=1+(t-1)m_{k-1}(n)+$

$+r-t$ удовлетворяет уравнениям

$$2^{m_{t-1}(n_1-1)}=2^{m_{k-1}(n)},\quad t+s_{t-1}(n_1-1)=r,$$

а следовательно, и уравнению

$$2^{m_{t-1}(n_1-1)}[t+s_{t-1}(n_1-1)]=r2^{m_{k-1}(n)}.$$

Поскольку (3.15) является следствием (3.13) и левая часть (3.15) возрастает по n_1, то n_1, определяемое соотношениями (3.12), (3.13), удовлетворяет неравенству

$$n_1\leqslant(t-1)m_{k-1}(n)+r-t. \tag{3.17}$$

В силу (3.17)

$$\begin{aligned}n-n_1-1&\geqslant n-(t-1)m_{k-1}(n)-r+t-1=\\&=n-\frac{t-1}{k-1}n+\frac{t-1}{k-1}s_{k-1}(n)-r+t-1=\\&=\frac{(k-t)n}{k-1}+\frac{t-1}{k-1}s_{k-1}(n)+t-r-1=\\&=(k-t)\left[m_{k-1}(n)+\frac{s_{k-1}(n)}{k-1}\right]+\frac{t-1}{k-1}s_{k-1}(n)+t-r-1=\\&=(k-t)m_{k-1}(n)+s_{k-1}(n)+t-r-1,\end{aligned}$$

откуда

$$\frac{n-n_1-1}{k-t}\geqslant m_{k-1}(n)+\frac{s_{k-1}(n)+t-r-1}{k-t} \tag{3.18}$$

и

$$\begin{aligned}m_{k-t}(n-n_1-1)-m_{k-1}(n)&\geqslant\\&\geqslant\frac{s_{k-1}(n)-s_{k-t}(n-n_1-1)+t-r-1}{k-t}.\end{aligned} \tag{3.19}$$

Учитывая, что, по определению x_1 и $s_{k-t}(n-n_1-1)$, соответственно $r\leqslant k+s_{k-1}(n)-1$ и $s_{k-t}(n-n_1-1)\leqslant k-t-1$, имеем из (3.19)

$$\begin{aligned}m_{k-t}(n-n_1-1)-m_{k-1}(n)&\geqslant\\&\geqslant\frac{s_{k-1}(n)-(k-t-1)+t-r-1}{k-t}\geqslant-2+\frac{1}{k-t}\end{aligned}$$

и, в силу целочисленности левой части, $m_{k-t}(n-n_1-1)-m_{k-1}(n)\geqslant-1$.

Пусть

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=-1; \tag{3.20}$$

тогда из (3.19)

$$s_{k-1}(n)-s_{k-t}(n-n_1-1)+t-r-1\leqslant t-k, \quad (3.21)$$

а доказываемое неравенство (3.14) имеет вид

$$2[k+s_{k-1}(n)-r]\leqslant k+1-t+s_{k-1}(n-1-n_1). \quad (3.22)$$

Заметим теперь, что в случае (3.20) из (3.18) следует, что $s_{k-1}(n)+t-r-1<0$, т. е. $s_{k-1}(n)+t-r\leqslant 0$, и используем (3.21), тогда (3.22) очевидно, так как

$$2[k+s_{k-1}(n)-r]-[k+1-t+s_{k-1}(n-1-n_1)]\equiv$$
$$\equiv[s_{k-1}(n)-s_{k-t}(n-n_1-1)+t-r-1]+k+s_{k-1}(n)-r\leqslant$$
$$\leqslant t-k+k+s_{k-1}(n)-r=s_{k-1}(n)+t-r\leqslant 0.$$

Пусть

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=0, \quad (3.23)$$

тогда, в силу (3.19), $s_{k-1}(n)-s_{k-t}(n-n_1-1)+t-r-1\leqslant$ $\leqslant 0$, а это и есть в случае (3.23) доказываемое неравенство (3.14).

Пусть, наконец,

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=q\geqslant 1. \quad (3.24)$$

Тогда из (3.19) получаем

$$s_{k-1}(n)-r\leqslant 1-t+s_{k-t}(n-n_1-1)+q(k-t).$$

Далее, $q(k-t)\leqslant(2^q-1)[k+1-t+s_{k-t}(n-n_1-1)]$, так как $q\leqslant 2^q-1$ при $q\geqslant 1$, и потому

$$k+s_{k-1}(n)-r\leqslant k+1-t+s_{k-t}(n-n_1-1)+q(k-t)\leqslant$$
$$\leqslant 2^q[k+1-t+s_{k-t}(n-n_1-1)].$$

Тем самым доказано неравенство (3.14) в предположении (3.24) и случай (3.16) полностью разобран.

Приступаем к рассмотрению случая

$$(t-1)/r\leqslant 1/2 \quad \text{или} \quad (t-1)/r\geqslant 1. \quad (3.25)$$

Докажем сначала два вспомогательных неравенства.

Во-первых, в предположениях (3.25)

$$(t-1)\log_2\frac{t-1}{r}\geqslant t-r-1. \quad (3.26)$$

Для доказательства положим $(t-1)/r=\tau$ и рассмотрим функцию $z(\tau)=\tau\log_2\tau-\tau+1$. Надо доказать, что $z(\tau)\geqslant 0$ при $0<\tau\leqslant 1/2$ или $\tau\geqslant 1$, но это очевидно, так

как функция $z(\tau)$ выпукла при $\tau > 0$ и $z(1/2) = z(1) = 0$.

Во-вторых,

$$(t-1)\log_2(t+\alpha) - \alpha \geqslant (t-1)\log_2(t-1) + 1 \quad \text{при} \quad 0 \leqslant \alpha \leqslant t-2. \tag{3.27}$$

Для доказательства рассмотрим вогнутую функцию $z(\alpha) = (t-1)\log_2(t+\alpha) - \alpha$ и убедимся в том, что $z(0) \geqslant z(t-2)$. Тогда $\min\limits_{0 \leqslant \alpha \leqslant t-2} z(\alpha) = z(t-2) = (t-1)\log_2(t-1) + 1$ и неравенство (3.27) доказано. Остается проверить, что $z(0) \geqslant z(t-2)$, т. е. $(t-1)\log_2[t/(t-1)] \geqslant 1$. Положим $\beta = 1/(t-1)$, тогда $0 \leqslant \beta \leqslant 1$ (так как, в силу (3.11), $t \geqslant 2$) и проверяемое неравенство принимает вид $\log_2(1+\beta) - \beta \geqslant 0$. Последнее при $0 \leqslant \beta \leqslant 1$ очевидно, так как $\log_2(1+\beta) - \beta$ — вогнутая функция, принимающая нулевые значения при $\beta = 0$, $\beta = 1$.

Произведем теперь некоторые эквивалентные преобразования над заданным нам неравенством (3.15). Имеем

$$m_{t-1}(n_1 - 1) + \log_2[t + s_{t-1}(n_1 - 1)] < m_{k-1}(n) + \log_2 r,$$

откуда

$$\frac{n_1 - 1}{t-1} + \frac{s_{t-1}(n_1 - 1)}{t-1} < \frac{n}{k-1} - \frac{s_{k-1}(n)}{k-1} + \log_2 \frac{r}{t + s_{t-1}(n_1 - 1)},$$

и после умножения обеих частей неравенства на $t-1$, вычитания из n и деления на $k-t$

$$\frac{n - n_1 - 1}{k-t} > \frac{n}{k-1} + \frac{(t-1)\log_2 \frac{t + s_{t-1}(n_1-1)}{r} - s_{t-1}(n_1 - 1) + \frac{t-1}{k-1} s_{k-1}(n) - 2}{k-t} =$$

$$= m_{k-1}(n) + \frac{(t-1)\log_2 \frac{t + s_{t-1}(n_1-1)}{r} - s_{t-1}(n_1-1) + s_{k-1}(n) - 2}{k-t}.$$

Воспользовавшись неравенствами (3.27) при $\alpha = s_{t-1}(n_1 - 1)$ и (3.26), получаем

$$m_{k-t}(n - n_1 - 1) - m_{k-1}(n) >$$

$$> \frac{(t-1)\log_2 \frac{t + s_{t-1}(n_1-1)}{r} - s_{t-1}(n_1 - 1) - s_{k-1}(n) - 2}{k-t} +$$

$$+\frac{-s_{k-t}(n-n_1-1)}{k-t}\geqslant$$

$$\geqslant\frac{(t-1)\log_2\frac{t-1}{r}-1+s_{k-1}(n)-s_{k-t}(n-n_1-1)}{k-t}\geqslant$$

$$\geqslant\frac{t-r-2+s_{k-1}(n)-s_{k-t}(n-n_1-1)}{k-t}. \quad (3.28)$$

Продолжая (3.28), имеем

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)>\frac{t-r-2+s_{k-1}(n)-(k-t-1)}{k-t}=$$

$$=-1+\frac{t-r-1+s_{k-1}(n)}{k-t}, \quad (3.29)$$

откуда, вспоминая, что $r\leqslant k+s_{k-1}(n)-1$, получаем неравенство

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)>-1+\frac{t-k}{k-t}=-2. \quad (3.30)$$

Пусть

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=-1. \quad (3.31)$$

Тогда в силу (3.28), $-1>[t-r-2+s_{k-1}(n)-s_{k-t}(n-n_1-1)]/(k-t)$ или

$$k-r-1+s_{k-1}(n)-s_{k-t}(n-n_1-1)\leqslant 0, \quad (3.32)$$

а в силу (3.29), $-1>-1+(t-r-1+s_{k-1}(n))/(k-t)$ или

$$t-r+s_{k-1}(n)\leqslant 0. \quad (3.33)$$

Складывая неравенства (3.32) и (3.33), получаем

$$2[k+s_{k-1}(n)-r]\leqslant k+1-t+s_{k-t}(n-n_1-1),$$

а это и есть в случае (3.31) доказываемое неравенство (3.14).

Пусть

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=0. \quad (3.34)$$

Тогда, в силу (3.28), $0>t-r-2+s_{k-1}(n)-s_{k-t}(n-n_1-1)$, следовательно, $k+s_{k-1}(n)-r\leqslant k+1-t+s_{k-t}(n-1-n_1)$, а это и есть в случае (3.34) доказываемое неравенство (3.14).

Пусть, наконец,

$$m_{k-t}(n-n_1-1)-m_{k-1}(n)=q\geqslant 1. \quad (3.35)$$

Тогда из (3.28) следует, что

$$q > \frac{t - r - 2 + s_{k-1}(n) - s_{k-t}(n - n_1 - 1)}{k - t},$$

и, стало быть,

$$s_{k-1}(n) - r \leqslant 1 - t + s_{k-t}(n - n_1 - 1) + q(k - t). \quad (3.36)$$

Далее, поскольку $q \leqslant 2^q - 1$ при $q \geqslant 1$, то

$$q(k - t) \leqslant (2^q - 1)[k + 1 - t + s_{k-t}(n - n_1 - 1)]. \quad (3.37)$$

Имеем, используя (3.36) и (3.37),

$$k + s_{k-1}(n) - r \leqslant k + 1 - t + s_{k-t}(n - n_1 - 1) + q(k - t) \leqslant$$
$$\leqslant 2^q[k + 1 - t + s_{k-t}(n - n_1 - 1)],$$

а это и есть в случае (3.35) доказываемое неравенство (3.14).

Согласно (3.30) все возможности исчерпаны и рассмотрение случая (3.25) закончено, а вместе с тем завершено рассмотрение случая (3.8).

Если же

$$b_k(n) \leqslant l \leqslant a_{k+1}(n), \quad k = 1, \ldots, [n/2], \quad (3.38)$$

то, выбирая на первом шаге $T_{n,r}(l) = r/(k + 1 + s_k(n))$, $r = 1, \ldots, k + s_k(n)$, минимизирующий игрок, как уже доказано при $l = a_{k+1}(n)$, может гарантировать себе проигрыш не более $W_n(l) = a_{k+1}(n)/(k + 1)$. При выполнении (3.38) проигрыш тем более не превосходит $W_n(l) = a_{k+1}(n)/(k + 1) = b_k(n)/k$.

Наконец, если

$$l \geqslant b_{[n/2]+1}, \quad (3.39)$$

то проигрыш не более $W_n(l) = \dfrac{b_{[n/2]} + 1}{[n/2] + 1} = \dfrac{1}{n + 1}$ минимизирующий игрок обеспечивает, выбирая на r-м шаге точку $T_{n,r}(l) = r/(n + 1)$, $r = 1, \ldots, [n/2] + 1 + s_{[n/2]+1}(n) = n$, независимо от действий максимизирующего игрока.

Рассмотрением возможностей $l \leqslant b_1(n)$, (3.8), (3.38) и (3.39) доказательство сформулированного в начале п. 2 утверждения закончено.

3. Доказательство теоремы 3.1 — гарантированный результат максимизирующего игрока. В этом пункте докажем, что, приписывая на первом шаге отрезкам $[0, x_1]$, $[x_1, 1]$ соответственно числа $l_1(x_1)$, $l - l_1(x_1)$, где $l_1(x_1)$

определяется формулой (3.5), максимизирующий игрок обеспечивает себе при оптимальных действиях в последующем выигрыш не менее (3.3).

Убедимся прежде всего, что (3.5) определяет стратегию максимизирующего игрока на первом шаге полностью (не обязательно однозначно).

Пусть

$$k-t+2 \leqslant s_k(n) \leqslant \frac{t-1}{2}-1, \quad t-1 \text{ четно}, \tag{3.40}$$

т. е. $l_1(x_1)$ не определено. Из второго двойного неравенства в (3.7) и левого — в (3.40) имеем $2t-s_k(n)-3 \leqslant k \leqslant t+s_k(n)-2$, откуда $t-1 \leqslant 2s_k(n)$, что противоречит правому неравенству (3.40), т. е. (3.40) невозможно.

Пусть

$$k-t+2 \leqslant s_k(n) \leqslant \frac{t}{2}-2, \quad t \text{ четно}, \tag{3.41}$$

т. е. $l_1(x_1)$ не определено. Тогда по-прежнему $t-1 \leqslant 2s_k(n)$, что противоречит правому неравенству (3.41), т. е. (3.41) невозможно.

Таким образом, при всех $x_1 \leqslant 1/2$ функция $l_1(x_1)$ определена первыми пятью строками (3.5), при $x_1 \geqslant 1/2$ функция $l_1(x_1)$ определена шестой строкой (3.5).

Так же легко проверяются условия $0 \leqslant l_1(x_1) \leqslant x_1$, $0 \leqslant l-l_1(x_1) \leqslant 1-x_1$, которым должна удовлетворять стратегия максимизирующего игрока.

Если $l \leqslant b_1(n)=(1/2)^n$, то максимизирующий игрок гарантирует выигрыш $W_n(l)=l$, приписывая на каждом шаге большему отрезку число l, меньшему — 0. Максимизирующий игрок, гарантируя выигрыш $b_1(n)$ при $l=b_1(n)$, тем более гарантирует $b_1(n)$ при $l \geqslant b_1(n)$. Первая строка в (3.5) этим полностью обоснована (заметим сразу, что шестая строка не требует дополнительного обоснования, если обоснованы предыдущие). Тем самым для $n=1$ доказано и утверждение п. 4.

Пусть утверждение доказано также для 2, ..., $(n-1)$-шаговых игр. Докажем его для n-шаговой игры.

Доказательство достаточно провести для случая (3.8), ибо если оно проведено для этого случая и $b_k(n) \leqslant l \leqslant a_{k+1}(n)$, $k=2, \ldots, [n/2]+1$, то максимизирующий игрок, гарантируя выигрыш $b_k(n)/k$ при $l=b_k(n)$, тем более гарантирует $b_k(n)/k$ при $l \geqslant b_k(n)$.

Итак, предположим, что в условиях (3.8) $x_1 \leqslant 1/(k+1+s_k(n))$. Согласно индукционному предположению, достаточно убедиться, что $(1-x_1)W_{n-1}(l/(1-x_1)) \geqslant W_n(l) = l/k$. Последнее неравенство, в свою очередь, достаточно проверить в наихудшем для максимизирующего игрока случае $x_1 = 1/(k+1+s_k(n))$:

$$\frac{k+s_k(n)}{k+1+s_k(n)} W_{n-1}\left(l\frac{k+1+s_k(n)}{k+s_k(n)}\right) \geqslant \frac{l}{k}.$$

Выписанное неравенство равносильно в силу определения функции W_{n-1} неравенству $l(k+1+s_k(n))/(k+s_k(n)) \leqslant b_k(n-1)$, для доказательства которого достаточно вследствие (3.8) показать, что $b_k(n)(k+1+s_k(n))/(k+s_k(n)) \leqslant b_k(n-1)$, или, что то же самое, $(k+s_k(n))2^{m_k(n)} \geqslant (k+1+s_k(n-1))2^{m_k(n-1)}$. Левая и правая части последнего соотношения связаны на самом деле знаком равенства, ибо если $m_k(n) = m_k(n-1)+1$, то $s_k(n-1) = k-1$, $s_k(n) = 0$, если же $m_k(n) = m_k(n-1)$, то $s_k(n) = s_k(n-1)+1$. Вторая строка в (3.5) обоснована.

Далее предстоит в условиях (3.8) рассмотреть случай (3.7). Мы покажем, что если

$$x_1 W_{n_1}\left(\frac{l_1(x_1)}{x_1}\right) < \frac{l}{k}, \tag{3.42}$$

то

$$(1-x_1)W_{n-n_1-1}\left(\frac{l-l_1(x_1)}{1-x_1}\right) \geqslant \frac{l}{k}. \tag{3.43}$$

Это, согласно индукционному предположению, означает следующее: если минимизирующий игрок выберет на отрезке $[0, x_1]$ такое число точек n_1 из оставшихся $n-1$ точек, что максимизирующий игрок не может гарантировать себе на отрезке $[0, x_1]$ выигрыш $W_n(l) = l/k$ — (3.42), то в этом случае максимизирующий игрок гарантирует себе выигрыш не менее $W_n(l) = l/k$ на отрезке $[x_1, 1]$ — (3.43), а стало быть, и во всей игре. Этот факт и требуется доказать.

Подставим в (3.42) $l_1(x_1) = (t-1)l/k$:

$$x_1 W_{n_1}\left(\frac{t-1}{x_1}\cdot\frac{l}{k}\right) < \frac{l}{k}.$$

Полученное неравенство равносильно, в силу определения

функции W_{n_1}, неравенству $\frac{t-1}{x_1}\cdot\frac{l}{k} > b_{t-1}(n_1)$, откуда, используя правую часть (3.9) и левую (3.7), имеем

$$(t-1)\left(\frac{1}{2}\right)^{m_k(n)-1}\frac{1}{k+1+s_k(n)}\cdot\frac{k+1+s_k(n)}{t-1} >$$
$$> \left(\frac{1}{2}\right)^{m_{t-1}(n_1)-1}\frac{t-1}{t+s_{t-1}(n_1)}.$$

После эквивалентных преобразований получаем последовательно

$$m_{t-1}(n_1) > m_k(n) + \log_2\frac{t-1}{t+s_{t-1}(n_1)},$$

$$n_1 - s_{t-1}(n_1) > (t-1)\,m_k(n) + (t-1)\log_2\frac{t-1}{t+s_{t-1}(n_1)}.$$

Далее, замечая, что $(t-1)/(t+s_{t-1}(n_1)) \geqslant 1/2$, продолжаем:

$$n_1 - s_{t-1}(n_1) > (t-1)\,m_k(n) + (t-1)\log_2\frac{1}{2} =$$
$$= (t-1)(m_k(n)-1),$$

и, вспоминая смысл входящих в неравенство величин, заключаем:

$$n_1 \geqslant (t-1)\,m_k(n). \tag{3.44}$$

Неравенство (3.44) получено как следствие (3.42). Подставляя в доказываемое неравенство (3.43) $l_1(x_1) = (t-1)\,l/k$ и заменяя его на эквивалентное, имеем последовательно

$$(1-x_1)\,W_{n-n_1-1}\left(\frac{k-t+1}{1-x_1}\cdot\frac{l}{k}\right) \geqslant \frac{l}{k},\quad \frac{k-t+1}{1-x_1}\cdot\frac{l}{k} \leqslant$$
$$\leqslant b_{k-t+1}(n-n_1-1).$$

Усилим полученное неравенство, используя правые неравенства (3.7), (3.9), вытекающее из (3.44) неравенство $n-n_1-1 \leqslant n-(t-1)\,m_k(n)-1$, а также убывание функции $b_k(n)$ по n. Получим

$$(k-t+1)\left(\frac{1}{2}\right)^{m_k(n)-1}\frac{1}{k+1+s_k(n)}\left(1-\frac{t}{k+1+s_k(n)}\right)^{-1} \leqslant$$
$$\leqslant \left(\frac{1}{2}\right)^{m_{k-t+1}(n-(t-1)m_k(n)-1)}\frac{k-t+1}{k-t+2+s_{k-t+1}(n-(t-1)m_k(n)-1)}$$

или

$$\left(\frac{1}{2}\right)^{m_k(n)}\frac{1}{k-t+1+s_k(n)}\leqslant\left(\frac{1}{2}\right)^{m_{k-t+1}(n-(t-1)m_k(n)-1)}\times$$
$$\times\frac{1}{k-t+2+s_{k-t+1}(n-(t-1)\,m_k(n)-1)}. \qquad (3.45)$$

Докажем, что на самом деле при $s_k(n)<k-t+1$ левая и правая части (3.45) связаны знаком равенства, откуда следует справедливость (3.43) и обоснованность третьей строки (3.5). Заметим, что

$$\frac{n-(t-1)\,m_k(n)-1}{k-t+1}=\frac{n-(t-1)\,(n/k-s_k(n)/k)-1}{k-t+1}=$$
$$=\frac{n}{k}+\frac{(t-1)\,\dfrac{s_k(n)}{k}-1}{k-t+1}=m_k(n)+\frac{s_k(n)}{k}+$$
$$+\frac{(t-1)\,s_k(n)}{k\,(k-t+1)}-\frac{1}{k-t+1}=m_k(n)+\frac{s_k(n)-1}{k-t+1}.$$

Если $s_k(n)=0$, то $m_{k-t+1}(n-(t-1)\,m_k(n)-1)=m_k(n)-1$, $s_{k-t+1}(n-(t-1)\,m_k(n)-1)=k-t$, если же $1\leqslant s_k(n)\leqslant k-t+1$, то $m_{k-t+1}(n-(t-1)\,m_k(n)-1)=m_k(n)$, $s_{k-t+1}(n-(t-1)\,m_k(n)-1)=s_k(n)-1$. Справедливость равенства в (3.45) в том и другом случае очевидна.

Случаи, соответствующие четвертой и пятой строкам (3.5), рассматриваются аналогично.

Подставим в (3.42) $l_1(x_1)=\frac{t-1}{2}\cdot\frac{l}{k}$. В качестве следствия полученного неравенства выводим

$$n_1>r\,[m_k(n)+1]+s_r(n_1)+r\log_2\frac{r}{r+1+s_r(n_1)}\geqslant$$
$$\geqslant rm_k(n)+s_r(n_1),$$

где $r=(t-1)/2$, $(t-1)$ четно, откуда $n_1\geqslant r[m_k(n)+1]$, и, стало быть, $n-n_1-1\leqslant n-\frac{t-1}{2}\,m_k(n)-\frac{t+1}{2}$. Заменяем неравенство (3.43) в рассматриваемом случае $l_1(x_1)=\frac{t-1}{2}\cdot\frac{l}{k}$ на эквивалентное неравенство и усиливаем его, используя последнее полученное неравенство

и правые неравенства (3.7), (3.9). Тогда имеем

$$\left(\frac{1}{2}\right)^{m_k(n)} \frac{1}{k-t+1+s_k(n)} \leqslant \left(\frac{1}{2}\right)^{m_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right)} \times$$

$$\times \frac{1}{k-\frac{t-3}{2}+s_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right)}. \qquad (3.46)$$

Замечаем далее, что

$$\frac{n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}}{k-\frac{t-1}{2}} = m_k(n) + \frac{s_k(n)-\frac{t+1}{2}}{k-\frac{t-1}{2}},$$

откуда выводим, что при четном $t-1$ и $s_k(n) \geqslant (t+1)/2$

$$m_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right) = m_k(n),$$

$$s_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right) = s_k(n)-\frac{t+1}{2}.$$

Если же $s_k(n) = (t-1)/2$, то

$$m_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right) = m_k(n)-1,$$

$$s_{k-\frac{t-1}{2}}\left(n-\frac{t-1}{2}m_k(n)-\frac{t+1}{2}\right) = k-\frac{t+1}{2}.$$

В обоих случаях левая и правая части (3.46) связаны знаком равенства, чем обоснована четвертая строка (3.5).

Подставим, наконец, в (3.42) $l_1(x_1) = \frac{t}{2}\cdot\frac{l}{k}$. В качестве следствия выводим

$$n_1 > r(m_k(n)+1) + s_r(n_1) + r\log_2 \frac{r-1/2}{r+1+s_r(n_1)},$$

где теперь $r = t/2$ (t четно). Если $s_r(n_1) \leqslant r-2$, то $\frac{r-1/2}{r+1+s_r(n_1)} \geqslant \frac{1}{2}$, и потому $n_1 - s_r(n_1) \geqslant rm_k(n)$, откуда

$$n \geqslant r(m_k(n)+1) \quad \text{при} \quad s_r(n_1) \leqslant r-2. \qquad (3.47)$$

Если же $s_r(n_1) = r-1$, то $r\log_2 \frac{r-1/2}{r+1+s_r(n_1)} = r\log_2 \frac{r-1/2}{2r} \geqslant -r-1$. Действительно, поделив обе ча-

сти неравенства на r и положив $\alpha = 1/(2r)$, $z(\alpha) = 2\alpha + \log_2(1-\alpha)$, перепишем неравенство в виде $z(\alpha) \geqslant 0$ при $0 \leqslant \alpha \leqslant 1/2$. Но последнее справедливо, так как $z(0) = z(1/2) = 0$ и функция z вогнута. Таким образом, $n_1 > r(m_k(n)+1) - 2$, откуда

$$n_1 \geqslant r(m_k(n)+1) - 1 \quad \text{при} \quad s_r(n_1) = r - 1. \tag{3.48}$$

Объединяя (3.47) и (3.48), имеем $n_1 \geqslant r(m_k(n)+1) - 1$ и, стало быть, $n - n_1 - 1 \leqslant n - r(m_k(n)+1)$. Преобразуя и усиливая теперь (3.43), получаем неравенство

$$\left(\frac{1}{2}\right)^{m_k(n)} \frac{1}{k+1-2r+s_k(n)} \leqslant \left(\frac{1}{2}\right)^{m_{k-r}(n-rm_k(n)-r)} \times$$
$$\times \frac{1}{k-r+1+s_{k-r}(n-rm_k(n)-r)}. \tag{3.49}$$

Замечаем далее, что

$$\frac{n-rm_k(n)-r}{k-r} = m_k(n) + \frac{s_k(n)-r}{k-r},$$

откуда выводим, что при четном t и $s_k(n) = r-1$

$$m_{k-r}(n-rm_k(n)-r) = m_k(n) - 1,$$
$$s_{k-r}(n-rm_k(n)-r) = k-r-1,$$

а при $s_k(n) \geqslant r$

$$m_{k-r}(n-rm_k(n)-r) = m_k(n),$$
$$s_{k-r}(n-rm_k(n)-r) = s_k(n) - r.$$

В обоих случаях, как нетрудно убедиться, левая и правая части (3.49) связаны знаком равенства, чем обоснована пятая строка (3.5).

Тем самым сформулированное в начале п. 3 утверждение доказано. Остается вернуться к последнему абзацу перед началом п. 2, чтобы убедиться, что теорема 3.1 полностью доказана. △

ГЛАВА 4

ПОИСК ГЛОБАЛЬНОГО ЭКСТРЕМУМА

В этой главе рассмотрены вопросы оптимального поиска наибольшего значения функций одной и нескольких переменных для классов функций, заданных квазиметриками, построены оптимальные детерминированные и стохастические пассивные алгоритмы, а также оптимальные на один шаг детерминированный и стохастический алгоритмы и последовательно-оптимальный детерминированный алгоритм.

Начнем с краткой исторической справки.

Первые работы по оптимальным (в рамках минимаксного подхода) алгоритмам поиска экстремума были посвящены унимодальным функциям, т. е. функциям с единственным максимумом. Ныне широко известна работа [371], где построен оптимальный последовательный алгоритм поиска экстремума унимодальной функции. Тот же результат получен в [364]. Последующими работами на эту тему были [4, 5, 128, 221, 264, 276, 291, 310, 311, 316—318, 347, 369, 372, 406, 422, 423, 466] и др. Некоторые из полученных результатов перенесены на многомерный случай [348, 370, 373—375, 403, 440, 462—465].

Вопросы построения оптимальных методов максимизации вогнутых (минимизации выпуклых) функций одной переменной обсуждаются в [4, 292, 436, 466]. Оптимальный последовательный алгоритм удалось построить лишь в случае, когда критерием является точность определения положения точки максимума [466]. В случае, когда критерием является точность определения максимального значения функции, удается построить лишь оптимальный на один шаг алгоритм [292].

Многие исследования посвящены оптимизации методов поиска максимума вогнутых (минимума выпуклых) функций нескольких переменных. Показано, что минимальное число информационных вычислений, позволяющее обеспечить заданную точность ε, равно по порядку $\ln(1/\varepsilon)$, как в случае, когда на каждом шаге вычисляется лишь значение максимизируемой функции, так и в слу-

чае, когда дополнительно вычисляется градиент (субградиент), причем нижняя по порядку оценка достигается, например, на методе центров тяжести [154] и на методе эллипсоидов [186].

Оптимальные алгоритмы для классов дифференцируемых функций изучаются в работах [57, 65, 109, 112, 119, 198, 295, 296] и др. В [112] показано, что наилучшая точность, достижимая при заданном числе информационных вычислений N, равна по порядку $N^{-(m+1)/n}$ для класса m раз дифференцируемых на $\mathbf{R}^n$ функций, удовлетворяющих условию

$$\left|\frac{\partial^m}{\partial t^m} f(x+te)\Big|_{t=t_1} - \frac{\partial^m}{\partial t^m} f(x+te)\Big|_{t=t_2}\right| \leqslant M\,|t_1-t_2|,$$

где $\|e\|=1$, а константа M не зависит от x, e. Этот результат справедлив как в случае, когда на каждом шаге вычисляются значения функции и всех ее частных производных до порядка m, так и в случае, когда вычисляются лишь значения самой функции. Ранее для классов m раз дифференцируемых функций аналогичные оценки были получены для других задач численного анализа [13, 52, 126]. В [112] построены также оптимальные по порядку алгоритмы для класса аналитических функций и некоторых других функциональных классов.

§ 1. О выборе начальных точек для методов локальной оптимизации

В вычислительной практике часто для отыскания глобального максимума функции f используют многократно тот или иной локальный метод. При этом начальные точки выбирают или случайно в соответствии с заданным вероятностным распределением, или детерминированно, чтобы они были в том или ином смысле равномерно распределены в $K\subset\mathbf{R}^n$ — области, на которой максимизируется функция f. Данный параграф посвящен вопросу оптимального выбора начальных точек.

1. Определения оптимального детерминированного и оптимального стохастического методов выбора начальных точек. Пусть надлежит выбрать N начальных точек $x_1, \ldots, x_N$. Естественно выбирать их так, чтобы минимизировать определенную формулой (2.2) гл. 2 величину $R(x^N)=\sup\limits_{x\in K}\min\limits_{i=1,\ldots,N}\rho(x, x_i)$ радиуса покрытия, за-

данного центрами $x^N = (x_1, \ldots, x_N)$ (здесь ρ — некоторая метрика). Такой выбор гарантирует, что расстояние от произвольной точки $x \in K$ до ближайшей из начальных точек не превосходит величины

$$R_N = \inf_{x_1, \ldots, x_N \in K} \sup_{x \in K} \min_{i=1, \ldots, N} \rho(x, x^i)$$

радиуса оптимального покрытия. Уменьшить это расстояние одновременно для всех точек из K невозможно.

Набор начальных точек, реализующий минимум в выражении для R_N, назовем *оптимальным детерминированным методом выбора* (*оптимальной детерминированной конфигурацией*) *начальных точек*. Таким образом, оптимальные детерминированные конфигурации совпадают с наборами центров оптимального покрытия множества K (по поводу оптимальных покрытий см. п. 1 § 2 гл. 2, а также п. 3 § 1 гл. 3).

Стохастические методы выбора начальных точек задаются вероятностными мерами $\sigma^N \in \Sigma^N$, где Σ^N — множество всех вероятностных мер на σ-алгебре борелевских подмножеств области K^N. Применяя стохастический метод σ^N, конфигурацию начальных точек $(x_1, \ldots, x_N)$ выбирают случайно в соответствии с вероятностной мерой σ^N.

Оптимальным стохастическим методом выбора начальных точек назовем метод, заданный вероятностной мерой, минимизирующей функцию

$$\overline{R}(\sigma^N) \stackrel{\text{def}}{=} \sup_{x \in K} \int_{K^N} \min_{i=1, \ldots, N} \rho(x, x_i)\, \sigma^N \{dx^N\}. \tag{1.1}$$

Таким образом, при применении оптимального стохастического метода расстояние в среднем от произвольной точки $x \in K$ до ближайшей из начальных точек не превосходит величины

$$\overline{R}_N \stackrel{\text{def}}{=} \inf_{\sigma^N \in \Sigma^N} \overline{R}(\sigma^N). \tag{1.2}$$

Уменьшить это расстояние в среднем для всех точек из K невозможно.

Сравнение стохастических и детерминированных методов выбора начальных точек проводится, например, в работах [301, 303], однако во всех рассмотренных случаях хотя бы один из сравниваемых методов не является оптимальным.

2. Сравнение оптимальных стохастических и детерминированных методов выбора начальных точек. Проведем сравнение для случая

$$K=\{u=(u^1,\ \ldots,\ u^n)\,|\,0\leqslant u^i\leqslant 1,\quad i=1,\ \ldots,\ n\},$$
$$\rho(u,v)=\max_{i=1,\ldots,n}|u^i-v^i|,\ N=p^n,\ p\text{ — целое число.} \tag{1.3}$$

Теорема 1.1 [240]. *В предположениях* (1.3) *справедливы неравенства*

$$\frac{n}{n+1}R_N\leqslant\overline{R}_N\leqslant R_N, \tag{1.4}$$

причем оценка снизу асимптотически точна, т. е.

$$\lim_{p\to\infty}\frac{\frac{n}{n+1}R_{p^n}}{\overline{R}_{p^n}}=1. \tag{1.5}$$

Доказательство. Справедливость правого из неравенств (1.4) очевидна, так как Σ^N содержит любую вероятностную меру, сосредоточенную в единственной точке множества K^N.

Для доказательства левого неравенства воспользуемся леммой 2.4 гл. 2. Учитывая, что в рассматриваемом случае $R_N=r_N=1/(2p)$, $\mu(K)=1$, имеем

$$\inf_{x^N\in K^N}\int_K\min_{i=1,\ldots,N}\rho(x,x_i)\,dx=\frac{n}{n+1}R_N,$$

откуда

$$\int_K\min_{i=1,\ldots,N}\rho(x,x_i)\,dx\geqslant\frac{n}{n+1}R_N,\quad x^N\in K^N. \tag{1.6}$$

Интегрируя обе части (1.6) по произвольной вероятностной мере $\sigma^N\in\Sigma^N$, получим неравенство

$$\int_K\left(\int_{K^N}\min_{i=1,\ldots,N}\rho(x,x_i)\,\sigma^N\{dx^N\}\right)dx\geqslant\frac{n}{n+1}R_N$$

(изменение порядка интегрирования законно вследствие непрерывности подынтегральной функции), из которого следует, что $\overline{R}(\sigma^N)\geqslant\frac{n}{n+1}R_N$. Отсюда в силу произвольности $\sigma^N\in\Sigma^N$ немедленно вытекает справедливость левого из неравенств (1.4).

Приступим к доказательству соотношения (1.5). Для этого введем в рассмотрение функцию

$$\varphi(n) = \frac{n^2 + \sqrt{3n^2 + 4n + 1} - 1}{2(n^2 + n)}, \quad n = 1, 2, \ldots$$

Нетрудно проверить, что $1/2 < \varphi(n) < 1/2 + 1/[2(n+1)]$.

Отметим, что всюду дальше $\varphi(n)$ можно заменить на $1/2$. При этом рассматриваемый ниже стохастический метод σ_*^N заменится на худший (доставляющий большее значение функции $\overline{R}$) метод, рассмотрение которого, однако, также обеспечивает доказательство соотношения (1.5). Усложнение конструкции вызвано желанием указать как можно более эффективный метод (оптимальный стохастический метод выбора начальных точек при $n > 1$ неизвестен).

Положим $\tau = 1/(p - 2 + 2\varphi(n))$ и рассмотрим куб

$$K(\tau) = \left\{u = (u^1, \ldots, u^n) \,\middle|\, -\frac{p\tau - 1}{2} \leqslant u^i \leqslant \right.$$
$$\left. \leqslant 1 + \frac{p\tau - 1}{2}, \ i = 1, \ldots, n\right\}$$

с ребрами длины $p\tau$. Каждое из ребер $K(\tau)$ разобьем на p частей и проведем гиперплоскости, параллельные координатным гиперплоскостям. Куб $K(\tau)$ разобьется при этом на $N = p^n$ кубов с ребрами длины τ. Центры этих кубов обозначим через $x_1^*, \ldots, x_N^*$. Пусть $x_*^N = (x_1^*, \ldots, x_N^*)$; θ — случайный вектор, равномерно распределенный в кубе $K_0 = \{\theta = (\theta^1, \ldots, \theta^n) \mid |\theta^i| \leqslant \tau/2, \ i = 1, \ldots, n\}$; $x_i^*(\theta)$ — проекция вектора $x^* + \theta$ на куб K, $i = 1, \ldots, N$; $x_*^N(\theta) = (x_1^*(\theta), \ldots, x_N^*(\theta))$. Вероятностную меру, соответствующую случайному вектору $x_*^N(\theta)$, обозначим через σ_*^N.

Вычислим величину

$$\overline{R}(\sigma_*^N) = \max_{a \in K} \int_{K^N} \min_{i=1,\ldots,N} \rho(a, x_i)\, \sigma_*^N\{dx^N\}.$$

На рис. 8 изображен случай $n = 2$, $p = 5$; K — это квадрат, обведенный жирной линией, $K(\tau)$ — больший квадрат.

Пусть $a = x_j^*$, причем x_j^* не является центром ни одного из граничных кубов. Имеем

$$\int\limits_{K^N} \min_{i=1,\ldots,N} \rho(x_j^*, x_i)\, \sigma_*^N \{dx^N\} = \frac{1}{\tau^n} \int\limits_{K_0} \rho(x_j^*, x_j^*(\theta))\, d\theta =$$

$$= \frac{1}{\tau^n} \int\limits_{K_0} \rho(0, \theta)\, d\theta = \frac{n}{n+1} \cdot \frac{\tau}{2}. \quad (1.7)$$

Пусть теперь $a = a_1$, причем координатный куб с ребрами длины τ и центром a_1 целиком лежит в K. Нетрудно видеть (см. рис. 8), что и в этом случае

$$\int\limits_{K^N} \min_{i=1,\ldots,N} \rho(a_1, x_i)\, \sigma_*^N \{dx^N\} = \frac{n}{n+1} \cdot \frac{\tau}{2}. \quad (1.8)$$

Пусть, наконец, точка $a = a_2$ такова, что координатный куб с ребрами длины τ и центром a_2 не лежит

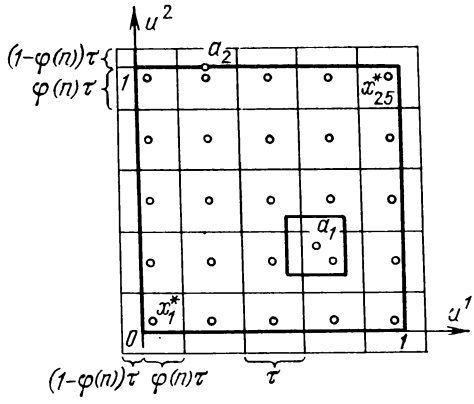

Рис. 8. К определению стохастического метода σ_*^N

целиком в K. Достаточно рассмотреть случай, когда a_2 лежит на границе K, причем все координаты a_2, кроме одной, больше 0 и меньше 1. Предположим без ограничения общности, что $a_2 = x_i^* + a'$, где $a' = (0, \ldots, 0, \tau/2 - (1 - \varphi(n))\tau)$. Такой случай изображен на рис. 8.

Имеем

$$\int\limits_{K^N} \min_{i=1,\ldots,N} \rho(a_2, x_i)\, \sigma_*^N\{dx^N\} =$$

$$= \int\limits_{\substack{|\theta^i| \leqslant \tau/2,\, i=1,\ldots,n-1 \\ -\tau/2 \leqslant \theta^n \leqslant \tau/2 + (1-\varphi(n))\tau}} \rho(\theta, a') \frac{1}{\tau^n}\, d\theta +$$

$$+ (1 - \varphi(n)) \int\limits_{K_0'} \rho'(0, \theta') \frac{1}{\tau^{n-1}}\, d\theta' = \frac{n}{n+1} \cdot \frac{\tau}{2}, \quad (1.9)$$

где

$$K_0' = \{\theta' = (\theta^1, \ldots, \theta^{n-1}) \,|\, |\theta^i| \leqslant \tau/2,\ i = 1, \ldots, n-1\},$$

$$\rho'(u', v') = \max_{i=1,\ldots,n-1} |u^i - v^i|, \quad u' = (u^1, \ldots, u^{n-1}),$$

$$v' = (v^1, \ldots, v^{n-1}).$$

Теперь на основании равенств (1.7)—(1.9) мы можем заключить, что

$$\overline{R}(\sigma_*^N) = \frac{n}{n+1} \cdot \frac{1}{2(p - 2 + 2\varphi(n))}. \quad (1.10)$$

Учитывая, что $R_{p^n} = 1/(2p)$, из (1.4) и (1.10) имеем

$$1 \geqslant \frac{\frac{n}{n+1} R_{p^n}}{\overline{R}_{p^n}} \geqslant \frac{\frac{n}{n+1} R_{p^n}}{\overline{R}\left(\sigma_*^{p^n}\right)} = \frac{p - 2 + 2\varphi(n)}{p}, \quad (1.11)$$

откуда непосредственно следует (1.5). △

Замечание 1. Нетрудно видеть, что

$$\frac{p - 2 + 2\varphi(n)}{p} = \frac{\frac{n}{n+1} R_{p^n}}{\overline{R}\left(\sigma_*^{p^n}\right)} \leqslant \frac{\overline{R}_{p^n}}{\overline{R}\left(\sigma_*^{p^n}\right)} \leqslant 1,$$

откуда следует, что стохастические методы $\sigma_*^{p^n}$ гарантируют при $p \to \infty$ последовательность результатов, эквивалентную оптимальной. △

Замечание 2. Теорема показывает, что в предположениях (1.3) относительный выигрыш от использования стохастических методов выбора начальных точек (по сравнению с детерминированными методами) невелик и сходит на нет с увеличением размерности пространства. △

§ 2. Оптимальный пассивный поиск для функционального класса, заданного квазиметрикой

В данном параграфе рассматривается вопрос об оптимальных детерминированных и стохастических пассивных алгоритмах для задачи поиска глобального экстремума (задачи глобальной оптимизации) функции, принадлежащей классу F_ρ, где ρ — произвольная квазиметрика. Здесь и в последующих параграфах данной главы предполагается, что $S(f) = \sup\limits_{x \in K} f(x)$, $B = \mathbf{R}$, а также, если явно не оговорено противное, что информационными вычислениями являются вычисления значений функции f и допустима единственная итоговая операция $\tilde{\beta}(x^N, y^N) = \max\limits_{i=1,\ldots,N} y_i$ (см. (3.6) гл. 1). При этом критерий (4.14) гл. 1 принимает вид

$$\varepsilon(x^N, f) = \sup_{x \in K} f(x) - \max_{i=1,\ldots,N} f(x_i) \tag{2.1}$$

(величина $\varepsilon(\tilde{x}^N, f)$ определяется второй из формул (4.14) гл. 1). Далее в этой главе мы будем вести речь о множествах алгоритмов X^N (см. (3.8) гл. 1) и $\tilde{X}^N$ (см. (3.10) гл. 1), не касаясь больше вопроса об итоговой операции.

1. Сведéние задачи построения оптимального детерминированного пассивного алгоритма к задаче оптимального покрытия множества K. Обозначим наилучший гарантированный в множестве алгоритмов X^N результат через v_N:

$$v_N = \inf_{x^N \in X^N} \sup_{f \in F_\rho} \varepsilon(x^N, f).$$

В формулировке следующей теоремы фигурируют величины $R(x^N)$ и R_N, определенные соответственно формулами (2.2) и (2.3) гл. 2.

Теорема 2.1 [240]. *Результат, гарантированный алгоритмом x^N на функциональном классе F_ρ, равен радиусу $R(x^N)$ покрытия, заданного центрами $x_1, \ldots, x_N$. Наилучший гарантированный результат v_N равен радиусу оптимального покрытия R_N. Алгоритм x_0^N оптимален по точности на классе F_ρ в множестве X^N тогда и только тогда, когда вектор x_0^N задает оптимальное покрытие множества K.*

Доказательство. Для функции h_a, введенной в лемме 2.1 гл. 1, имеем

$$\varepsilon(x^N, h_a) = \sup_{x\in K} h_a(x) - \max_{i=1,\ldots,N} h_a(x_i) = \min_{i=1,\ldots,N} \rho(x_i, a). \tag{2.2}$$

Пусть $f \in F_\rho$, $\delta > 0$, $f(x_\delta) \geqslant \sup_{x\in K} f(x) - \delta$. Тогда

$$\varepsilon\left(x^N, h_{x_\delta}\right) = \min_{i=1,\ldots,N} \rho(x_i, x_\delta) = \rho\left(x_{i_\delta}, x_\delta\right) \geqslant$$
$$\geqslant f(x_\delta) - f\left(x_{i_\delta}\right) \geqslant f(x_\delta) - \max_{i=1,\ldots,N} f(x_i) \geqslant$$
$$\geqslant \sup_{x\in K} f(x) - \max_{i=1,\ldots,N} f(x_i) - \delta = \varepsilon\left(x^N, f\right) - \delta. \tag{2.3}$$

Следовательно, $\sup_{a\in K} \varepsilon\left(x^N, h_a\right) \geqslant \varepsilon\left(x^N, f\right) - \delta$. Учитывая произвольность $f \in F_\rho$ и устремляя δ к нулю, выводим отсюда, что

$$\sup_{a\in K} \varepsilon\left(x^N, h_a\right) \geqslant \sup_{f\in F_\rho} \varepsilon\left(x^N, f\right).$$

Справедливость противоположного неравенства вытекает из леммы 2.1 гл. 1, поэтому с учетом (2.2) получаем

$$\sup_{f\in F_\rho} \varepsilon\left(x^N, f\right) = \sup_{a\in K} \varepsilon\left(x^N, h_a\right) = \sup_{a\in K} \min_{i=1,\ldots,N} \rho(x_i, a) = R\left(x^N\right).$$

Остальные утверждения теоремы теперь очевидны. △

Замечание 1. Из доказательства теоремы видно, что если $\bar{F} \subset F_\rho$ и $h_a \in \bar{F}$ при всех $a \in K$, то алгоритм x_0^N, оптимальный на классе F_ρ, оптимален также и на подклассе $\bar{F}$ класса F_ρ. Наилучшие гарантированные результаты на F_ρ и на $\bar{F}$ совпадают. △

Замечание 2. Оптимальные на классе F_ρ пассивные детерминированные алгоритмы для задач восстановления функции и глобальной оптимизации совпадают. В обоих случаях дело сводится к отысканию оптимального покрытия множества K. Таким образом, так же, как и в задаче восстановления функции из класса F_ρ по ее значениям, в случае $n = 1$, $K = [0, 1]$, $\rho(u, v) = M|u - v|$ алгоритм $(1/(2N), 3/(2N), \ldots, (2N-1)/(2N))$ оптимален в множестве всех пассивных алгоритмов, при этом $v_N = M/(2N)$ (см. следствие из теоремы 1.2 гл. 3). Как мы убедимся ниже, оптимальные пассивные стохастиче-

ские алгоритмы и последовательно-оптимальные алгоритмы для задач восстановления функции и глобальной оптимизации существенно различны. △

Замечание 3. Напомним, что для класса F_ρ оптимальный пассивный алгоритм оптимален также в множестве всех последовательных алгоритмов (следствие из теоремы 5.3 гл. 1). Этот факт можно доказать и иным методом [240], аналогичным методу доказательства теоремы 1.3 гл. 3. △

2. Сведéние задачи построения оптимального стохастического пассивного алгоритма к решению антагонистической игры. В силу определения (7.8) гл. 1, для отыскания оптимального по точности стохастического алгоритма необходимо минимизировать по $\sigma^N \in \Sigma^N$ функцию $\sup\limits_{f\in F_\rho} \int\limits_{K^N} \varepsilon(x^N, f)\,\sigma^N\{dx^N\}$. Здесь Σ^N — по-прежнему множество всех вероятностных мер на σ-алгебре борелевских подмножеств множества K^N (предполагается, что K — открытое множество, либо замыкание открытого множества).

Лемма 2.1. *Пусть K — компакт, функция ρ обладает свойствами квазиметрики и непрерывна на $K\times K$. Тогда для любого $\sigma^N \in \Sigma^N$*

$$\sup_{f\in F_\rho} \int\limits_{K^N} \varepsilon(x^N, f)\,\sigma^N\{dx^N\} = \sup_{a\in K} \int\limits_{K^N} \varepsilon(x^N, h_a)\,\sigma^N\{dx^N\} =$$
$$= \sup_{a\in K} \int\limits_{K^N} \min_{i=1,\ldots,N} \rho(x_i, a)\,\sigma^N\{dx^N\}.$$

Доказательство. Пусть $f\in F_\rho$. В силу непрерывности ρ функция f также непрерывна. Следовательно, существует точка $x_0\in K$ такая, что $f(x_0) = \max\limits_{x\in K} f(x)$. Из (2.3) вытекает, что для произвольного $x^N \in K^N$ имеем $\varepsilon(x^N, f) \leqslant \varepsilon(x^N, h_{x_0})$. Отсюда

$$\int\limits_{K^N} \varepsilon(x^N, f)\,\sigma^N\{dx^N\} \leqslant \int\limits_{K^N} \varepsilon(x^N, h_{x_0})\,\sigma^N\{dx^N\},$$

и потому

$$\sup_{f\in F_\rho} \int\limits_{K^N} \varepsilon(x^N, f)\,\sigma^N\{dx^N\} \leqslant \sup_{a\in K} \int\limits_{K^N} \varepsilon(x^N, h_a)\,\sigma^N\{dx^N\}.$$

Противоположное неравенство очевидно, так как $h_a \in F_\rho$ при $a \in K$. Тем самым справедливость первого равенства утверждения леммы доказана. Справедливость второго следует из (2.2). △

Следствие. *Оптимальный по точности стохастический пассивный алгоритм глобальной оптимизации совпадает с оптимальной смешанной стратегией минимизирующего игрока в антагонистической игре с функцией выигрыша* $\min\limits_{i=1,\ldots,N} \rho(x_i, a)$. *Чистыми стратегиями минимизирующего игрока в этой игре являются векторы* $x^N \in K^N$, *чистыми стратегиями максимизирующего игрока — точки* $a \in K$. *Наилучший гарантированный в множестве стохастических пассивных алгоритмов результат равен значению игры.*

3. Построение оптимального стохастического пассивного алгоритма в одномерном случае. В одномерном случае удается получить полное решение задачи об оптимальном по точности стохастическом пассивном алгоритме, который, оказывается, имеет весьма простой вид.

Теорема 2.2 [240]. *Пусть* $n = 1$, $K = [0, 1]$, $\rho(u, v) = M|u - v|$. *Тогда стохастический пассивный алгоритм* σ_0^N, *приписывающий вероятности* 1/2 *векторам*

$$x_1^N = \left(0, \frac{2}{2N-1}, \ldots, \frac{2N-2}{2N-1}\right), \quad x_2^N = \left(\frac{1}{2N-1}, \frac{3}{2N-1}, \ldots, 1\right),$$

оптимален по точности на классе F_ρ; *наилучший гарантированный результат равен* $M/(4N-2)$.

Доказательство. В соответствии со следствием из леммы 2.1 достаточно показать, что σ_0^N — оптимальная смешанная стратегия минимизирующего игрока в игре с функцией выигрыша

$$\eta(x^N, a) \stackrel{\text{def}}{=} \min_{i=1,\ldots,N} |x_i - a|,$$

а $s_N = 1/(4N-2)$ — ее значение. Покажем сначала, что стратегия σ_0^N гарантирует минимизирующему игроку результат s_N.

Пусть $a \in K = [0, 1]$. Тогда $a = a' + j/(2N-1)$, $j \in \{0, 1, \ldots, 2N-1\}$, $0 \leqslant a' \leqslant 1/(2N-1)$ и

$$\bar{\eta}(\sigma_0^N, a) = \frac{1}{2}\eta(x_1^N, a) + \frac{1}{2}\eta(x_2^N, a) = \frac{1}{2}a' + \frac{1}{2}\left(\frac{1}{2N-1} - a'\right) = s_N,$$

где

$$\bar{\eta}(\sigma^N, a) = \int\limits_{K^N} \eta(x^N, a)\, \sigma^N\{dx^N\}, \quad \sigma^N \in \Sigma^N.$$

Таким образом, стратегия σ_0^N действительно гарантирует результат s_N.

Пусть ω_0 — смешанная стратегия максимизирующего игрока, приписывающая $2N$ точкам 0, $1/(2N-1)$, $2/(2N-1)$, ..., $(2N-1)/(2N-1)=1$ равные вероятности $1/(2N)$. Покажем, что ω_0 гарантирует максимизирующему игроку результат не хуже s_N.

Пусть $x^N=(x_1, \ldots, x_i, \ldots, x_N)\in K^N$ и $j/(2N-1)<x_i<(j+1)/(2N-1)$ для некоторого целого j. Тогда имеет место хотя бы одно из двух неравенств

$$\begin{aligned} &\bar{\eta}\left(\left(x_1, \ldots, \frac{j}{2N-1}, \ldots, x_N\right), \omega_0\right) \leqslant \bar{\eta}(x^N, \omega_0), \\ &\bar{\eta}\left(\left(x_1, \ldots, \frac{j+1}{2N-1}, \ldots, x_N\right), \omega_0\right) \leqslant \bar{\eta}(x^N, \omega_0), \end{aligned} \tag{2.4}$$

где

$$\bar{\eta}(x^N, \omega) = \int\limits_K \eta(x^N, a)\, \omega\{da\}, \quad \omega \in \Sigma,$$

Σ — множество всех вероятностных мер на σ-алгебре борелевых подмножеств множества K. Действительно,

$$\bar{\eta}(x^N, \omega_0) = \frac{1}{2N} \sum_{\alpha=0}^{2N-1} \eta\left(x^N, \frac{\alpha}{2N-1}\right) = \\ = \frac{1}{2N} \sum_{\alpha=0}^{2N-1} \min_{\beta=1,\ldots,N} \left| x_\beta - \frac{\alpha}{2N-1} \right|.$$

Положим

$$M(i) = \left\{ \frac{m}{2N-1} \,\middle|\, \left| x_i - \frac{m}{2N-1} \right| = \min_{j=1,\ldots,N} \left| x_j - \frac{m}{2N-1} \right|, \right. \\ \left. 0 \leqslant m \leqslant 2N-1 \right\},$$

т. е. $M(i)$ — все те точки из множества $\left\{\frac{0}{2N-1}, \frac{1}{2N-1}, \ldots, \frac{2N-1}{2N-1}\right\}$, для которых x_i — ближайшая точка из множества точек $\{x_1, \ldots, x_N\}$. Тогда, если слева от x_i точек

из $M(i)$ больше, чем справа, то справедливо первое из неравенств (2.4), в противном случае — второе. Если слева и справа от x_i точек из $M(i)$ поровну, то справедливы оба неравенства (2.4).

Проведя такие же рассуждения для всех $i=1, \ldots, N$, получим неравенство

$$\bar{\eta}\left(\left(\frac{j_1}{2N-1}, \ldots, \frac{j_N}{2N-1}\right), \omega_0\right) \leqslant \bar{\eta}(x^N, \omega_0),$$

где $j_1, \ldots, j_N \in \{0, 1, \ldots, 2N-1\}$. Наконец,

$$\bar{\eta}\left(\left(\frac{j_1}{2N-1}, \ldots, \frac{j_N}{2N-1}\right), \omega_0\right) =$$

$$= \frac{1}{2N} \sum_{\alpha=0}^{2N-1} \min_{\beta=1,\ldots,N} \left| \frac{j_\beta}{2N-1} - \frac{\alpha}{2N-1} \right| \geqslant \frac{1}{2N} \cdot \frac{N}{2N-1} = s_N.$$

В силу произвольности $x^N \in K^N$ из последних двух неравенств следует, что стратегия ω_0 гарантирует результат не хуже s_N. △

Замечание 1. Результат $M/(4N-2)$, гарантированный оптимальным по точности стохастическим пассивным алгоритмом, почти в два раза лучше результата $M/(2N)$, гарантированного оптимальным по точности детерминированным пассивным алгоритмом. △

Замечание 2. В задаче восстановления функции по ее значениям стохастический пассивный алгоритм σ_0^N не является оптимальным по точности. Чтобы убедиться в этом, достаточно рассмотреть функцию $f(x) = \min\{x, |x-2/3|\}$ на отрезке $K=[0, 1]$ при $N=2$. Как нетрудно проверить, для нее $\int\limits_{K^N} \varepsilon(x^N, f)\, \sigma_0^N\{dx^N\} = M/3$ (здесь $\varepsilon(x^N, f)$ определяется формулой (1.8) гл. 3), и, стало быть, алгоритм σ_0^N гарантирует результат не лучше $M/3$. В то же время оптимальный по точности детерминированный пассивный алгоритм $x_0^N = (1/4, 3/4)$ гарантирует результат $M/4$. △

4. Сравнение в многомерном случае наилучших гарантированных результатов в множествах детерминированных и стохастических пассивных алгоритмов.

Теорема 2.3. *Пусть* $K = \{u = (u^1, \ldots, u^n) \mid 0 \leqslant u^i \leqslant 1,\ i=1, \ldots, n\}$, $\rho(u, v) = M \max\limits_{i=1,\ldots,n} |u^i - v^i|$, $N = p^n$, *p — целое число. Тогда наилучшие гарантированные ре-*

зультаты в множествах K^N и Σ^N — соответственно детерминированных пассивных и стохастических пассивных алгоритмов — связаны между собой следующим соотношением:

$$\frac{n}{n+1} \min_{x^N \in K^N} \max_{f \in F_\rho} \varepsilon\left(x^N, f\right) \leqslant$$
$$\leqslant \min_{\sigma^N \in \Sigma^N} \max_{f \in F_\rho} \int_{K^N} \varepsilon\left(x^N, f\right) \sigma^N\left\{dx^N\right\} \leqslant \min_{x^N \in K^N} \max_{f \in F_\rho} \varepsilon\left(x^N, f\right),$$

причем оценка снизу асимптотически точна, т. е.

$$\lim_{p \to \infty} \frac{\dfrac{n}{n+1} \min\limits_{x^{p^n} \in K^{p^n}} \max\limits_{f \in F_\rho} \varepsilon\left(x^{p^n}, f\right)}{\min\limits_{\sigma^{p^n} \in \Sigma^{p^n}} \max\limits_{f \in F_\rho} \int\limits_{K^{p^n}} \varepsilon\left(x^{p^n}, f\right) \sigma^{p^n}\left\{dx^{p^n}\right\}} = 1.$$

Доказательство. Учитывая, что функция ρ в (1.3) и функция ρ, фигурирующая в формулировке доказываемой теоремы, различаются лишь постоянным множителем M, и принимая во внимание теорему 2.1 и лемму 2.1, видим, что наша теорема есть просто сформулированная в иных терминах теорема 1.1. △

Замечание 1. Построенные при доказательстве теоремы 1.1 стохастические пассивные алгоритмы (в терминах теоремы 1.1 — стохастические методы) $\sigma_*^{p^n}$ гарантируют при $p \to \infty$ последовательность результатов, эквивалентную оптимальной (см. замечание 1 к теореме 1.1):

$$\max_{f \in F_\rho} \int_{K^{p^n}} \varepsilon\left(x^{p^n}, f\right) \sigma_*^{p^n}\left\{dx^{p^n}\right\} \sim$$
$$\sim \min_{\sigma^{p^n} \in \Sigma^{p^n}} \max_{f \in F_\rho} \int_{K^{p^n}} \varepsilon\left(x^{p^n}, f\right) \sigma^{p^n}\left\{dx^{p^n}\right\}. \ \triangle$$

Замечание 2. Теорема 2.3 показывает, что в данной ситуации относительный выигрыш от использования стохастических алгоритмов (по сравнению с детерминированными алгоритмами) невелик. Он выражается коэффициентом $n/(n+1)$, стремящимся к единице при стремлении размерности пространства к бесконечности (ср. с замечанием 2 к теореме 1.1). △

§ 3. Сведение задачи построения последовательно-оптимального алгоритма для функционального класса, заданного квазиметрикой, к серии задач оптимального покрытия

В этом параграфе мы выясняем структуру последовательно-оптимальных алгоритмов. Именно, показано, что построение последовательно-оптимального на классе F_ρ алгоритма сводится к решению серии специального вида задач оптимального покрытия. Рассматривая функциональный класс F_ρ, будем предполагать, что K — компакт и квазиметрика ρ непрерывна на множестве $K \times K$, не упоминая это каждый раз в формулировках лемм. Если не оговорено противное, считаем, что $F = F_\rho$. В частности,

$$F(z^i) = \{f \in F_\rho \mid f(x_j) = y_j, \quad j = 1, \ldots, i\},$$

причем во всех последующих построениях ситуация z^i предполагается реализуемой, т. е. $F(z^i) \neq \varnothing$.

1. Вспомогательные утверждения. Положим

$$\bar{y}_i = \max_{j=1,\ldots,i} y_j,$$

$$K_i(t) = \{x \mid \varphi_{2i}(x) \leqslant t\} =$$

$$= \bigcup_{j=1}^{i} \{x \mid y_j + \rho(x, x_j) \leqslant t\} = \bigcup_{j=1}^{i} S(x_j, t - y_j),$$

$$h_i(x) = \min\{\varphi_{2i}(x), \bar{y}_i\},$$

где мажоранта φ_{2i} для функций класса $F(z^i)$ определена формулой (1.3) гл. 2, а $S(v, R) = \{u \mid \rho(u, v) \leqslant R\}$ есть ρ-шар радиуса R с центром v (если $R < 0$, то $S(v, R) = \varnothing$)[1]. Кроме того, будем пользоваться обозначениями: $CA = K \backslash A$ для $A \subset K$, $\bar{A}$ — замыкание множества A. Нетрудно видеть, что

$$h_i(x) = \begin{cases} \varphi_{2i}(x), & x \in K_i(\bar{y}_i), \\ \bar{y}_i, & x \in CK_i(\bar{y}_i). \end{cases}$$

В силу леммы 2.7 гл. 1 $h_i \in F(z^i)$. Пусть $a \in \overline{CK_i(\bar{y}_i + r)} \neq \varnothing$, $r \geqslant 0$. Тогда

$$\rho(a, x_j) \geqslant \bar{y}_i - y_j + r, \quad j = 1, \ldots, i. \tag{3.1}$$

---

[1]) Так же как и для ранее введенных функций φ_{1i}, φ_{2i}, среди аргументов и индексов отображения $K_i(t)$ для упрощения обозначений не фигурирует явным образом $z^i = (x_1, \ldots, x_i, y_1, \ldots, y_i)$. О зависимости от z^i напоминает индекс i. То же самое относится к вводимым ниже функциям $h_i(x)$, $g_i(x)$, $R_i(x_{i+1}, \ldots, x_N)$, R_{iN}.

Отметим, что в точках множества $\overline{CK_i(\bar{y}_i+r)}$ и только в точках этого множества функции из $F(z^i)$ могут принимать значения $y \geqslant \bar{y}_i + r$. Положим $g_i(x) = \max\{h_i(x), \bar{y}_i + r - \rho(a, x)\}$.

Лемма 3.1. *Справедливы соотношения*

$$g_i(x) = \begin{cases} \bar{y}_i + r - \rho(a, x), & \rho(a, x) \leqslant r, \\ h_i(x), & \rho(a, x) \geqslant r, \end{cases} \quad g_i \in F(z^i).$$

Доказательство. При $\rho(a, x) \leqslant r$ имеем $\bar{y}_i + r - \rho(a, x) \geqslant \bar{y}_i \geqslant h_i(x)$, т. е. $g_i(x) = \bar{y}_i + r - \rho(a, x)$. Если $\rho(a, x) \geqslant r$, то

$$\bar{y}_i - (\bar{y}_i + r - \rho(a, x)) \geqslant 0, \tag{3.2}$$

$$\varphi_{2i}(x) - (\bar{y}_i + r - \rho(a, x)) =$$
$$= \min_{j=1,\ldots,i} \{y_j + \rho(x, x_j)\} - (\bar{y}_i + r - \rho(a, x)) =$$
$$= y_s + \rho(x, x_s) - (\bar{y}_i + r - \rho(a, x)) \geqslant y_s - \bar{y}_i - r + \rho(a, x_s) \geqslant 0, \tag{3.3}$$

где последнее неравенство справедливо в силу (3.1), а предпоследнее — в силу свойств квазиметрики (2.3), (2.4) гл. 1. Из (3.2) и (3.3) следует, что $h_i(x) \geqslant \bar{y}_i + r - \rho(a, x_i)$, т. е. $g_i(x) = h_i(x)$.

Наконец, вследствие леммы 2.7 гл. 1, $g_i \in F$. Кроме того, $g_i(x_j) = h_i(x_j) = y_j$, так как $\rho(a, x_j) \geqslant r$, $j = 1, \ldots, i$ (см. (3.1)). Таким образом, $g_i \in F(z^i)$. $\triangle$

Положим

$$R_i(x_{i+1}, \ldots, x_N) = \min\left\{r \mid r \geqslant 0, \bigcup_{j=i+1}^{N} S(x_j, r) \supset \overline{CK_i(\bar{y}_i + r)}\right\}.$$

Убедиться в существовании числа $R_i(x_{i+1}, \ldots, x_N)$ и заодно найти выражение для него можно следующим образом:

$$R_i(x_{i+1}, \ldots, x_N) =$$
$$= \min\left\{r \mid r \geqslant 0, \bigcup_{j=i+1}^{N} S(x_j, r) \cup K(\bar{y}_i + r) \supset K\right\} =$$
$$= \min\left\{r \mid r \geqslant 0, \bigcup_{j=i+1}^{N} S(x_j, r) \cup \left[\bigcup_{j=1}^{i} S(x_j, \bar{y}_i - y_j + r)\right] \supset K\right\} =$$
$$= \min\{r \mid r \geqslant 0, \sup_{x \in K} \min\Big[\min_{j=i+1,\ldots,N} \rho(x, x_j),$$

$$\min_{j=1,\ldots,i} (\rho(x, x_j) - \bar{y}_i + y_j)] \leqslant r\} =$$

$$= \sup_{x\in K} \min \left\{ \min_{j=i+1,\ldots,N} \rho(x, x_j), \min_{j=1,\ldots,i} (\rho(x, x_j) - \bar{y}_i + y_j) \right\}. \tag{3.4}$$

Заметим, что последнее выражение неотрицательно, так как для $x = x_{i_0}$ в силу реализуемости z^i справедливы неравенства $\rho(x, x_j) - \bar{y}_i + y_j \geqslant 0$, $j = 1, \ldots, i$, где $i_0 \in \{1, \ldots, i\}$ определяется из условия $y_{i_0} = \bar{y}_i$.

Докажем теперь следующий почти очевидный факт.

Лемма 3.2. *Пусть вектор* $(x_{i+1}, \ldots, x_N)$ *задает покрытие множества* $A \subset K$ *радиуса* r, *т. е.*

$$\sup_{x\in A} \min_{j=i+1,\ldots,N} \rho(x, x_j) = r. \tag{3.5}$$

Тогда существует такая точка $a \in \bar{A}$, *что* $\min_{j=i+1,\ldots,N} \rho(x_j, a) = r$.

Доказательство. В силу (3.5) для произвольного целого положительного k существует такая точка $a_k \in A$, что $\min_{j=i+1,\ldots,N} \rho(x_j, a_k) > r - 1/k$. Последовательность a_k имеет предельную точку $a \in \bar{A}$. Для нее

$$\rho(x_j, a) \geqslant \rho(x_j, a_k) - \rho(a, a_k) > r - 1/k - \rho(a, a_k),$$
$$j = i+1, \ldots, N,$$

откуда в силу произвольности k выводим, что $\min_{j=i+1,\ldots,N} \rho(x_j, a) \geqslant r$. Справедливо и противоположное неравенство, т. к. радиус покрытия с центрами $x_1, \ldots, x_N$ множества $\bar{A}$, так же как и множества A, равен r. △

Следующий результат является аналогом для класса $F(z^i)$ результата теоремы 2.1, при этом эффективность пассивных алгоритмов $(x_{i+1}, \ldots, x_N)$ оценивается по критерию $\varepsilon(x^i, x_{i+1}, \ldots, x_N, f)$.

Лемма 3.3. *Результат, гарантированный пассивным алгоритмом* $(x_{i+1}, \ldots, x_N)$ *на подклассе* $F(z^i)$ *класса* F_ρ, *равен величине* $R_i(x_{i+1}, \ldots, x_N)$, *т. е.*

$$\sup_{f\in F(z^i)} \varepsilon(x^i, x_{i+1}, \ldots, x_N, f) = R_i(x_{i+1}, \ldots, x_N).$$

Доказательство. Пусть

$$f \in F(z^i), \max_{x\in K} f(x) = f(x_0), \quad x_0 \in K, \quad r \geqslant R_i(x_{i+1}, \ldots, x_N).$$

Если $x_0 \in K_i(\bar{y}_i + r)$, то $x_0 \in S(x_j, \bar{y}_i - y_j + r)$, $j \in \{1, \ldots, i\}$, и, следовательно, $f(x_0) \leqslant f(x_j) + \rho(x_0, x_j) \leqslant y_j + (\bar{y}_i - y_j + r) = \bar{y}_i + r$. Стало быть,

$$\varepsilon(x^i, x_{i+1}, \ldots, x_N, f) = f(x_0) - \max\{\bar{y}_i, f(x_{i+1}), \ldots, f(x_N)\} \leqslant \leqslant f(x_0) - \bar{y}_i \leqslant r.$$

Если же $x_0 \in CK_i(\bar{y}_i + r)$, то $x_0 \in S(x_j, r)$, $j \in \{i+1, \ldots, N\}$, и, следовательно,

$$\varepsilon(x^i, x_{i+1}, \ldots, x_N, f) \leqslant f(x_0) - f(x_j) \leqslant \rho(x_0, x_j) \leqslant r.$$

Итак, в любом случае $\varepsilon(x^i, x_{i+1}, \ldots, x_N, f) \leqslant r$. Используя произвольность функции $f \in F(z^i)$ и числа $r \geqslant R_i(x_{i+1}, \ldots, x_N)$, получаем, что

$$\sup_{f \in F(z^i)} \varepsilon(x^i, x_{i+1}, \ldots, x_N, f) \leqslant R_i(x_{i+1}, \ldots, x_N). \tag{3.6}$$

Пусть теперь $r < R_i(x_{i+1}, \ldots, x_N)$. Тогда радиус покрытия с центрами $x_{i+1}, \ldots, x_N$ множества $CK_i(\bar{y}_i + r)$ больше r, и вследствие леммы 3.2 существует такая точка $a \in \overline{CK_i(\bar{y}_i + r)}$, что

$$\min_{j=i+1,\ldots,N} \rho(x_j, a) > r. \tag{3.7}$$

Для выбранных a и r рассмотрим функцию g_i, определенную перед формулировкой леммы 3.1. В силу (3.7) и леммы 3.1

$$\varepsilon(x^i, x_{i+1}, \ldots, x_N, g_i) = = g_i(a) - \max\{\bar{y}_i, g_i(x_{i+1}), \ldots, g_i(x_N)\} = \bar{y}_i + r - \bar{y}_i = r, \tag{3.8}$$

откуда, учитывая произвольность $r < R_i(x_{i+1}, \ldots, x_N)$, получаем

$$\sup_{f \in F(z^i)} \varepsilon(x^i, x_{i+1}, \ldots, x_N, f) \geqslant R_i(x_{i+1}, \ldots, x_N). \tag{3.9}$$

Неравенства (3.6) и (3.9) доказывают лемму.

2. Последовательно-оптимальный по точности алгоритм. Положим

$$R_{iN} = \min_{x_{i+1}, \ldots, x_N \in K} R_i(x_{i+1}, \ldots, x_N). \tag{3.10}$$

Существование минимума в правой части (3.10) следует из компактности множества K и непрерывности функции $R_i(x_{i+1}, \ldots, x_N)$ по совокупности переменных $x_{i+1}, \ldots, x_N$

(непрерывность легко установить на основе равенства (3.4)). Можно сказать, что R_{iN} есть минимальное из чисел r таких, что радиус оптимального покрытия с $N-i$ центрами множества $CK_i(\bar{y}_i+r)$ не превосходит r.

Теорема 3.1 [241]. *Предположим, что K — компакт, функция ρ обладает свойствами квазиметрики и непрерывна на множестве $K\times K$. Пусть $\tilde{x}_1^0$ — произвольный из центров оптимального покрытия с N центрами множества K, $\tilde{x}_{i+1}^0(z^i)$ — произвольный из центров любого такого покрытия с $N-i$ центрами множества $CK_i(\bar{y}_i+R_{iN})$, что радиус покрытия не превосходит R_{iN}, $i=1, \ldots, N-1$ (иными словами, $\tilde{x}_{i+1}^0(z^i)$ — одна из точек, реализующих минимум в* (3.10)).

Тогда алгоритм $\tilde{x}_0^N=(\tilde{x}_1^0, \ldots, \tilde{x}_N^0)$ последовательно-оптимален по точности на классе F_ρ и гарантирует после i шагов в ситуации z^i результат R_{iN}.

Доказательство. Правомерность выбора последовательно-оптимальным алгоритмом точки $\tilde{x}_1^0$, указанной в формулировке теоремы, следует из того, что последовательно-оптимальный алгоритм оптимален в $\tilde{X}^N$ (структура оптимального в $\tilde{X}^N$ алгоритма устанавливается теоремой 2.1 и замечанием 3 к ней).

По построению алгоритм $\tilde{x}_0^N$ гарантирует после i шагов в ситуации z^i результат R_{iN}. В соответствии с определением последовательно-оптимального алгоритма (см. (6.6) гл. 1) для доказательства теоремы достаточно показать, что

$$\inf_{\tilde{x}_{i+1}, \ldots, \tilde{x}_N} \sup_{f\in F(z^i)} \varepsilon(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f)=R_{iN}. \quad (3.11)$$

Пусть $\tilde{x}_{i+1}, \ldots, \tilde{x}_N$ — произвольные отображения вида (3.1) гл. 1, $r<R_{iN}$, а вектор $(x^i, x_{i+1}, \ldots, x_N)$ получается при применении (см. формулы (3.2) гл. 1) алгоритма $(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N)$ к функции h_i, определенной в начале п. 1. Тогда $r<R_i(x_{i+1}, \ldots, x_N)$. Определим точку a и функцию g_i так же, как в доказательстве леммы 3.3. Заметим, что $g_i(x_j)=h_i(x_j)$, $j=1, \ldots, N$, поэтому при применении алгоритма $(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N)$ к функции g_i получается тот же вектор $(x^i, x_{i+1}, \ldots, x_N)$, что и при применении к функции h_i. Следовательно, с учетом (3.8),

$$\varepsilon(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, g_i)=\varepsilon(x^i, x_{i+1}, \ldots, x_N, g_i)=r,$$

откуда в силу произвольности $\tilde{x}_{i+1}, \ldots, \tilde{x}_N$

$$\inf_{\tilde{x}_{i+1},\ldots,\tilde{x}_N} \sup_{f\in F(z^i)} \varepsilon\left(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f\right) \geqslant r,$$

а в силу произвольности $r < R_{iN}$

$$\inf_{\tilde{x}_{i+1},\ldots,\tilde{x}_N} \sup_{f\in F(z^i)} \varepsilon\left(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f\right) \geqslant R_{iN}. \quad (3.12)$$

С другой стороны, из леммы 3.3 немедленно получаем равенство

$$\inf_{x_{i+1},\ldots,x_N\in K} \sup_{f\in F(z^i)} \varepsilon\left(x^i, x_{i+1}, \ldots, x_N, f\right) = R_{iN}. \quad (3.13)$$

Далее, очевидно, что

$$\inf_{\tilde{x}_{i+1},\ldots,\tilde{x}_N} \sup_{f\in F(z^i)} \varepsilon\left(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_N, f\right) \leqslant$$
$$\leqslant \inf_{x_{i+1},\ldots,x_N\in K} \sup_{f\in F} \varepsilon\left(x^i, x_{i+1}, \ldots, x_N, f\right),$$

откуда, учитывая (3.13), выводим неравенство, противоположное неравенству (3.12). Тем самым соотношение (3.11) доказано. △

Замечание. Нетрудно убедиться, что при любом y_1 точка $\tilde{x}_2^0(z^1)$ является одним из $N-1$ центров (отличных от центра $\tilde{x}_1^0$) оптимального покрытия с N центрами множества K. Кроме того, если $y_{i+1} = \bar{y}_i$, то точка $\tilde{x}_{i+2}^0\left(z^{i+1}\right)$ является одним из $N-i-1$ центров (отличных от центра $\tilde{x}_{i+1}^0(z^i)$) оптимального покрытия с $N-i$ центрами, построенного при отыскании точки $\tilde{x}_{i+1}^0(z^i)$. В этом случае наилучшие гарантированные результаты после i шагов и после $i+1$ шагов совпадают. Можно сказать, что случай $y_{i+1} = \bar{y}_i$ есть случай наихудшего для вычислителя «поведения» функции f на $(i+1)$-м шаге. △

Теорема 3.1 сводит задачу построения последовательно-оптимального алгоритма к совокупности N специального вида задач оптимального покрытия. Решения этих задач для некоторых частных случаев даны в § 4.

3. Последовательно-оптимальный по числу информационных вычислений алгоритм. Предположим теперь, что фиксирована точность ε, которую необходимо обеспечить при решении задачи. Построим последовательно-оптимальный по числу информационных вычислений

алгоритм поиска глобального экстремума, проводя при этом все необходимые доказательства.

За точку первого информационного вычисления следует принять произвольный центр любого покрытия K минимальным числом ρ-шаров радиуса ε. Это — очевидное следствие определения последовательно-оптимального алгоритма (см. (6.10) гл. 1).

Пусть i информационных вычислений произведено. В силу (3.11)

$$\inf_{\tilde{x}_{i+1},\ldots,\tilde{x}_{i+s}} \sup_{f\in F(z^i)} \varepsilon\left(x^i, \tilde{x}_{i+1}, \ldots, \tilde{x}_{i+s}, f\right) = R_{i,i+s}.$$

Поэтому, учитывая (6.8) гл. 1 и (3.10), определенное формулой (6.9) гл. 1 число $N_\varepsilon(z^i)$ можно представить в виде

$$\begin{aligned} N_\varepsilon\left(z^i\right) &= \min\{s \mid R_{i,i+s} \leqslant \varepsilon\} = \\ &= \min\{s \mid R_i(x_{i+1}, \ldots, x_{i+s}) \leqslant \varepsilon \\ &\quad \text{для некоторых } x_{i+1}, \ldots, x_{i+s} \in K\} = \\ &= \min\left\{s \,\middle|\, \bigcup_{j=i+1}^{i+s} S(x_j, \varepsilon) \supset CK_i\left(\bar{y}_i + \varepsilon\right) \right. \\ &\qquad \left. \text{для некоторых } x_{i+1}, \ldots, x_{i+s} \in K\right\}. \end{aligned}$$

Если $x_{i+1}, \ldots, x_{i+N_\varepsilon(z^i)}$ таковы, что $\bigcup_{j=i+1}^{i+N_\varepsilon(z^i)} S(x_j, \varepsilon) \supset$ $\supset CK_i\left(\bar{y}_i + \varepsilon\right)$, то $R_i\left(x_{i+1}, \ldots, x_{i+N_\varepsilon(z^i)}\right) \leqslant \varepsilon$, и в силу леммы 3.3

$$\sup_{f\in F(z^i)} \varepsilon\left(x^i, x_{i+1}, \ldots, x_{i+N_\varepsilon(z^i)}, f\right) \leqslant \varepsilon. \tag{3.14}$$

Итак, для отыскания точки $(i+1)$-го информационного вычисления достаточно построить множество $CK_i(\bar{y}_i + \varepsilon)$ и найти минимальное s такое, что s ρ-шарами радиуса ε можно покрыть множество $CK_i(\bar{y}_i + \varepsilon)$. Неравенство (3.14) показывает, что за точку $(i+1)$-го информационного вычисления можно принять произвольный центр любого такого покрытия.

Если $CK_i(\bar{y}_i + \varepsilon) = \varnothing$, то задача решена с требуемой точностью ε и вычисления следует прекратить.

Заметим, что получившаяся задача проще, чем в случае фиксированного N. В рассматриваемом случае не нужно на $(i+1)$-м шаге решать оптимизационную задачу на отыскание R_{iN}. Роль этого числа играет при всех i постоянная ε.

§ 4. Конкретные вычислительные алгоритмы

Здесь результаты предыдущего параграфа конкретизируются применительно к одномерным задачам. Кроме того, показано, как использовать в многомерном случае, когда непосредственная реализация последовательно-оптимального алгоритма затруднена, знание его структуры.

1. Последовательно-оптимальный по точности алгоритм для класса функций одной переменной, удовлетворяющих условию Липшица. Предположим, что число N информационных вычислений фиксировано, $n=1$, $K=[0, 1]$, $\rho(u, v)=M|u-v|$. При этом $S(x, r)=[x-r/M, x+r/M]$ есть сегмент длины $2r/M$. Согласно теореме 3.1,

$$x_1 \in \left\{\frac{1}{2N}, \frac{3}{2N}, \ldots, \frac{2N-1}{2N}\right\}, \tag{4.1}$$

причем любая из точек (4.1) может быть выбрана последовательно-оптимальным алгоритмом в качестве точки первого информационного вычисления.

Пусть произведено i вычислений в точках $x_1, \ldots, x_i$; $y_j=f(x_j)$, $j=1, \ldots, i$; $z^i=(x^i, y^i)$; $x_{i1}, \ldots, x_{ii}$ — перестановка чисел $x_1, \ldots, x_i$ в порядке возрастания, $y_{ij}=f(x_{ij})$, β_j и α_{j+1} — соответственно меньший и больший корни уравнения $y_{ij}+M|x-x_{ij}|=\bar{y}_i$, т. е.

$$\beta_j = x_{ij} - \frac{\bar{y}_i - y_{ij}}{M}, \quad \alpha_{j+1} = x_{ij} + \frac{\bar{y}_i - y_{ij}}{M}, \quad j=1, \ldots, i$$

(см. рис. 9, где изображен случай $i=4$). Тогда

$$CK_i(\bar{y}_i) = [0, \beta_1) \cup \left[\bigcup_{j=2}^{i} (\alpha_j, \beta_j)\right] \cup (\alpha_{i+1}, 1]$$

(это множество на рис. 9 выделено жирной линией),

$$CK_i(\bar{y}_i + r) = \left[0, \beta_1 - \frac{r}{M}\right) \cup \left[\bigcup_{j=2}^{i} \left(\alpha_j + \frac{r}{M}, \beta_j - \frac{r}{M}\right)\right] \cup$$
$$\cup \left(\alpha_{i+1} + \frac{r}{M}, 1\right]).$$

При этом $[0, \beta) \stackrel{\text{def}}{=} \varnothing$, если $\beta \leqslant 0$; $(\alpha, \beta) \stackrel{\text{def}}{=} \varnothing$, если $\alpha \geqslant \beta$; $(\alpha, 1] \stackrel{\text{def}}{=} \varnothing$, если $\alpha \geqslant 1$. Введем еще обозначение $l_j = \beta_j - \alpha_j$, $j = 1, \ldots, m$.

Минимальное число сегментов длины $2r/M$, которыми можно покрыть множество $[0, \beta_1 - r/M)$ равно, очевидно, $\lceil \max\{(\beta_1 - r/M)/(2r/M), 0\}\rceil$. Аналогично, это

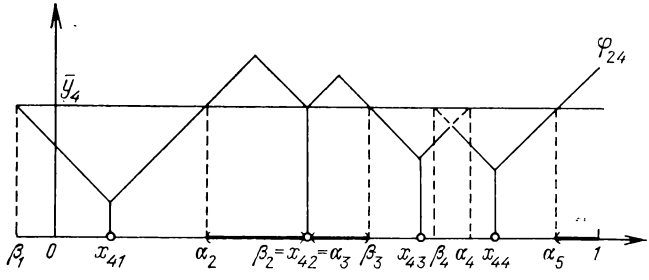

Рис. 9. К вопросу о построении последовательно-оптимального по точности алгоритма

число равно $\lceil \max\{(l_j - 2r/M)/(2r/M), 0\}\rceil$ для множества $(\alpha_j + r/M, \beta_j - r/M)$ и $\lceil \max\{(1 - \alpha_{i+1} - r/M)/(2r/M), 0\}\rceil$ — для множества $(\alpha_{i+1} + r/M, 1]$. Поэтому неравенство $r \geqslant R_{iN}$ эквивалентно неравенству

$$\left\lceil \max\left\{\frac{\beta_1 - r/M}{2r/M}, 0\right\}\right\rceil + \sum_{j=2}^{i}\left\lceil \max\left\{\frac{l_j/2 - r/M}{r/M}, 0\right\}\right\rceil + \\ + \left\lceil \max\left\{\frac{1 - \alpha_{i+1} - r/M}{2r/M}, 0\right\}\right\rceil \leqslant N - i, \quad (4.2)$$

и, следовательно,

$$R_{iN} = \min\left\{r \mid r > 0, \left\lceil \max\left\{\frac{\beta_1 - r/M}{2r/M}, 0\right\}\right\rceil + \right. \\ \left. + \sum_{j=2}^{i}\left\lceil \max\left\{\frac{l_j/2 - r/M}{r/M}, 0\right\}\right\rceil + \left\lceil \max\left\{\frac{1 - \alpha_{i+1} - r/M}{2r/M}, 0\right\}\right\rceil \leqslant N - i\right\}. \\ (4.3)$$

Алгоритм отыскания числа R_{iN} основан на формуле (4.3) и следующем соображении: хотя бы одно из чисел

$$\frac{\beta_1 - r/M}{2r/M}, \quad \frac{l_2 - r/M}{r/M}, \quad \ldots, \quad \frac{l_i/2 - r/M}{r/M}, \quad \frac{1 - \alpha_{i+1} - r/M}{2r/M} \quad (4.4)$$

при $r = R_{iN}$ является целым неотрицательным. В про-

тивном случае нашлось бы меньшее значение $r > 0$, при котором неравенство (4.2) еще выполнялось бы. Алгоритм состоит из двух этапов.

1°. Находятся все такие r, при которых величины (4.4) принимают целые значения $N - i$, $N - i - 1$, ..., 1, 0. Все найденные значения r нумеруются в порядке возрастания: $r_1 < r_2 < \ldots < r_s$, где $s \leqslant (i+1) \times \times (N - i + 1)$.

2°. В порядке возрастания индекса t проверяется выполнение неравенства (4.2) при $r = r_t$. Если это неравенство впервые окажется выполненным при $t = t_0$, то $R_{iN} = r_{t_0}$.

После того как R_{iN} найдено, нетрудно указать возможные в соответствии с теоремой 3.1 для последовательно-оптимального алгоритма точки $(i+1)$-го информационного вычисления $(i = 1, \ldots, N-1)$:

$$x_{i+1} \in \{(2k_1 - 1) R_{iN}/M \mid (2k_1 - 1) R_{iN}/M < \beta_1\} \cup$$
$$\cup \left[\bigcup_{j=2}^{i} \{\alpha_j + 2k_j R_{iN}/M \mid \alpha_j + 2k_j R_{iN}/M < \beta_j\} \right] \cup$$
$$\cup \{1 - (2k_{i+1} - 1) R_{iN}/M \mid 1 - (2k_{i+1} - 1) R_{iN}/M > \alpha_{i+1}\}. \tag{4.5}$$

Числа $k_1, \ldots, k_{i+1}$ в формуле (4.5) принимают целые положительные значения, причем любая точка из указанного набора может быть выбрана последовательно-оптимальным алгоритмом в качестве точки $(i+1)$-го информационного вычисления.

Может случиться, что множество (4.5) окажется пустым. Это означает, что оставшимися $N - i$ вычислениями нельзя при наихудшем «поведении» функции улучшить гарантированный результат, поэтому в качестве x_{i+1} можно выбирать любую точку из $K = [0, 1]$. Однако целесообразно этот выбор проводить в соответствии с правилом

$$x_{i+1} \in \{(2k_1 - 1) R_{iN}/M \mid (2k_1 - 1) R_{iN}/M \leqslant \beta_1\} \cup$$
$$\cup \left[\bigcup_{j=2}^{i} \{\alpha_j + 2k_j R_{iN}/M \mid \alpha_j + 2k_j R_{iN}/M \leqslant \beta_j\} \right] \cup$$
$$\cup \{1 - (2k_{i+1} - 1) R_{iN}/M \mid 1 - (2k_{i+1} - 1) R_{iN}/M \geqslant \alpha_{i+1}\}. \tag{4.6}$$

Отметим, что способы покрытия множества $CK_i(\bar{y}_i + R_{iN})$ «слева направо», на котором основаны

первые две строки формулы (4.5), и «справа налево», на котором основана третья строка, не исчерпывают множества способов, удовлетворяющих условиям теоремы 3.1. Поэтому формула (4.5) задает, вообще говоря, не все множество последовательно-оптимальных алгоритмов.

Можно, например, «симметризовать» способы покрытия, соответствующие формулам (4.5), (4.6), следующим образом. Пусть

$$n_1^i = \left[\max\left\{\frac{\beta_1 - R_{iN}/M}{2R_{iN}/M}, 0\right\}\right], \quad n_j^i = \left[\max\left\{\frac{l_j/2 - R_{iN}/M}{R_{iN}/M}, 0\right\}\right],$$

$$j = 2, \ldots, i, \quad n_{i+1}^i = \left[\max\left\{\frac{1 - \alpha_{i+1} - R_{iN}/M}{2R_{iN}/M}, 0\right\}\right]. \quad (4.7)$$

Нетрудно видеть, что алгоритм, для которого

$$x_{i+1} \in \left\{(2k_1 - 1)\frac{\beta_1}{2n_1^i + 1} \,\middle|\, k_1 = 1, \ldots, n_1^i\right\} \cup$$
$$\cup \left[\bigcup_{j=2}^{i} \left\{\alpha_j + k_j \frac{\beta_j - \alpha_j}{n_j^i + 1} \,\middle|\, k_j = 1, \ldots, n_j^i\right\}\right] \cup$$
$$\cup \left\{\alpha_{i+1} + 2k_{i+1}\frac{1 - \alpha_{i+1}}{2n_{i+1}^i + 1} \,\middle|\, k_{i+1} = 1, \ldots, n_{i+1}^i\right\}, \quad (4.8)$$

последовательно-оптимален по точности (если $n_j^i = 0$, то предполагается, что соответствующий набор пуст).

2. Оптимальный на один шаг алгоритм для класса функций одной переменной, удовлетворяющих условию Липшица. В соответствии с (4.1) при $N = 1$

$$x_1 = 1/2. \quad (4.9)$$

Точку x_{i+1} выберем в соответствии с вариантом (4.7), (4.8) последовательно-оптимального алгоритма (при этом в (4.3) следует полагать $N - i = 1$). Нетрудно видеть, что

$$n_r^i = 1; \quad n_j^i = 0, \quad j \neq r, \quad (4.10)$$

если максимальным среди чисел β_1, $l_1/2$, ..., $l_i/2$, $1 - \alpha_{i+1}$ является r-е число. Если максимальное число не единственно, то $n_1^i = \ldots = n_{i+1}^i = 0$ (этот случай соответствует случаю, когда множество (4.5) пусто), и в качестве x_{i+1} можно выбирать любую точку из $K = [0, 1]$. Тем не менее мы и в этом случае оставим в силе согла-

шение (4.10), предполагая, что r — любой, например, наименьший из номеров максимальных чисел. Из результатов предыдущего пункта непосредственно следует, что алгоритм (4.8)—(4.10) оптимален на один шаг. Отметим, что формулу (4.8) в случае (4.10) можно записать в виде

$$x_{i+1} = \begin{cases} \arg\max\limits_{x\in K} \varphi_{2i}(x), & \arg\max\limits_{x\in K} \varphi_{2i}(x) \notin \{0, 1\}, \\ \beta_1/3, & \arg\max\limits_{x\in K} \varphi_{2i}(x) = 0, \\ 1-(1-\alpha_{i+1})/3, & \arg\max\limits_{x\in K} \varphi_{2i}(x) = 1, \end{cases}$$

где $\varphi_{2i}(x) = \min\limits_{j=1,\ldots,i} \{y_j + M|x - x_j|\}$. Отсюда ясно, что оптимальный на один шаг алгоритм (4.8)—(4.10), начиная с третьего шага, совпадает с алгоритмом

$$x_1 = 0, \quad x_{i+1} = \arg\max_{x\in K} \varphi_{2i}(x), \quad i \geqslant 1, \tag{4.11}$$

называемым иногда *методом ломаных* и изучавшимся во многих работах [48, 88, 89, 153, 199, 201, 265, 427, 428] (при $n > 1$ оптимальный на один шаг алгоритм не определяется условием $x_{i+1} = \arg\max\limits_{x\in K} \varphi_{2i}(x)$).

3. Последовательно-оптимальный по числу информационных вычислений алгоритм для класса функций одной переменной, удовлетворяющих условию Липшица. Пусть по-прежнему $n = 1$, $K = [0, 1]$, $\rho(u, v) = M|u - v|$, но в отличие от п. 1 фиксирована точность ε, которую необходимо обеспечить, произведя минимальное число информационных вычислений. Построим в соответствии с п. 3 § 3 алгоритм, последовательно-оптимальный по числу информационных вычислений.

Точка x_1 определяется формулой (4.1) при $N = N_\varepsilon = \lceil M/(2\varepsilon)\rceil$ (при отыскании x_1 можно исходить и из других способов покрытия сегмента $[0, 1]$ минимальным числом сегментов длины $2\varepsilon/M$).

Пусть i информационных вычислений произведено. Тогда

$$CK_i(\bar{y}_i + \varepsilon) = \left[0, \beta_1 - \frac{\varepsilon}{M}\right) \cup \left[\bigcup_{j=2}^{i} \left(\alpha_j + \frac{\varepsilon}{M}, \beta_j - \frac{\varepsilon}{M}\right)\right] \cup$$
$$\cup \left(\alpha_{i+1} + \frac{\varepsilon}{M}, 1\right].$$

Если это множество пусто, то задача решена с требуемой точностью и вычисления прекращаются. В противном случае точку x_{i+1} можно выбирать либо в соответствии с формулой (4.5) при $R_{iN} = \varepsilon$, либо в соответствии с формулами (4.7), (4.8) при $R_{iN} = \varepsilon$.

Результаты проведенных численных экспериментов показывают, что чем выше требуемая точность решения задачи, т. е. чем меньше ε, тем заметнее преимущество последовательно-оптимального над оптимальным алгоритмом (правильнее говорить о модифицированных последовательно-оптимальном и оптимальном алгоритмах, поскольку вычисления производились с использованием адаптивной оценки константы M по способу (4.1) гл. 2). Так, для функции $f(x) = 3\cos 2x - 2\sin x$, $x \in K = [-1, 1]$, такой вывод можно сделать из табл. 5. В ней

Таблица 5

ε	0.1	0.01	0.001
N_ε	15	82	269
N	66	980	9803
N/N_ε	0.227	0.084	0.027
$\varepsilon_{\text{факт}}$	0.7×10^{-5}	0.2×10^{-7}	0.4×10^{-8}

$N_\varepsilon = [M(b-a)/(2\varepsilon)]$ — число информационных вычислений, при котором оптимальный алгоритм гарантирует достижение точности ε (при этом M — это оценка константы Липшица, полученная способом (4.1) гл. 2 по окончании информационных вычислений); N — число информационных вычислений, при котором последовательно-оптимальный алгоритм обеспечивает достижение той же точности ε; $\varepsilon_{\text{факт}}$ — истинная точность решения задачи последовательно-оптимальным алгоритмом.

Как видим, преимущество последовательно-оптимального алгоритма над оптимальным в задаче глобальной оптимизации выражено ярче, чем в задачах численного интегрирования и восстановления функции. Это и понятно, так как в задачах интегрирования и восстановления необходимо так или иначе исследовать поведение функции на всей области определения, в то время как в задаче глобальной оптимизации поиск достаточно ве-

сти лишь в «перспективной», с точки зрения уже полученной информации, области.

Обращает на себя внимание и то, что истинная точность решения задачи оказывается на несколько порядков лучше заданной точности. Такое соотношение характерно для многих задач. Оно связано с тем, что для функции, которая хорошо «ведет себя» в окрестности экстремума, оценка константы Липшица для этой окрестности оказывается сильно завышенной. Данное обстоятельство следует учитывать при решении практических задач.

Во всех тестах при достаточно малом ε выявляется также преимущество последовательно-оптимального над оптимальным на один шаг алгоритмом (в качестве тестовых функций были выбраны начальные отрезки ряда Фурье со случайно генерируемыми коэффициентами).

4. О возможности использования единой схемы построения последовательно-оптимальных алгоритмов. Описанная схема построения последовательно-оптимальных алгоритмов поиска экстремума функций одной переменной применима не только для класса F_ρ в случае, когда информация состоит в вычислении значений функции и $\rho(u, v) = M|u - v|$, но и для других моделей вычислений. Как и в задаче численного интегрирования, эти модели должны удовлетворять условию (3.37) гл. 2. Еще раз зафиксируем основные моменты применимой в этом случае схемы построения последовательно-оптимальных алгоритмов.

На $(i+1)$-м шаге определяется перспективная для дальнейшего поиска область $\{x|\varphi_{2i}(x) > \bar{y}_i\}$, где, как и ранее, φ_{2i} — мажоранта для функций из определяемого информацией z^i подкласса $F(z^i)$ рассматриваемого функционального класса F. Пусть замыкание этой области состоит из промежутков

$$[0, \beta_1], [\alpha_2, \beta_2], \ldots, [\alpha_i, \beta_i], [\alpha_{i+1}, 1]. \qquad (4.12)$$

Далее отыскивается оптимальное распределение вычислительных ресурсов по промежуткам (4.12) и при найденном распределении строятся оптимальные алгоритмы для функциональных классов, соответствующих этим промежуткам. За x_{i+1} принимается точка первого информационного вычисления любого из построенных алгоритмов. Таким образом, задача построения последовательно-оптимального алгоритма оказывается сведен-

ной к серии задач построения оптимальных алгоритмов и целочисленных задач распределения ресурсов.

Существенно проще задача отыскания оптимального на один шаг алгоритма. В ряде случаев, решая ее, оказывается возможным построить эффективный вычислительный алгоритм, в то время как задача отыскания последовательно-оптимального алгоритма не поддается решению. В качестве примера можно привести оптимальный на один шаг алгоритм минимизации выпуклых функций [292]. Этим же способом построен эффективный алгоритм максимизации функций с ограниченной на отрезке вещественной прямой кусочно-непрерывной второй производной в случае, когда информационными вычислениями являются вычисления значений функции и ее первой производной в точках отрезка [296].

5. Построение оптимального по числу информационных вычислений алгоритма в двумерном случае. Вернемся к рассмотрению класса $F = F_\rho$. Пусть $n = 2$, $K = \{(u^1, u^2) \mid 0 \leqslant u^i \leqslant c^i,\ i = 1, 2\}$, $\rho(u, v) = M \max\{|u^1 - v^1|, |u^2 - v^2|\}$ и необходимо построить последовательно-оптимальный по числу информационных вычислений алгоритм, обеспечивающий точность ε. Положим $r = \varepsilon/M$, $l = 2r$. Множество $CK_i(\bar{y}_i + \varepsilon)$ получается выбрасыванием из K координатных квадратов со сторонами длины не меньше l (ρ-шаров радиуса, не меньше ε). В качестве точки $(i+1)$-го информационного вычисления последовательно-оптимальный алгоритм указывает любой из центров какого-либо покрытия этого множества наименьшим числом координатных квадратов со сторонами длины l.

Положим $x_1 = (r, r)$, $x_2 = (3r, r)$. Пусть вычисления в точках $x_1, \ldots, x_i$ произведены, $y_j = f(x_j)$, $j = 1, \ldots, i$. Выберем такое натуральное s, что прямая $u^2 = c^2 - sl$ имеет непустое пересечение с множеством $CK_i(\bar{y}_i + \varepsilon)$, а пересечение прямой $u^2 = c^2 - (s+1)l$ с этим множеством пусто. Осуществим покрытие одномерного множества

$$\{u = (u^1, u^2) \mid u \in CK_i(\bar{y}_i + \varepsilon),\ u^2 = c^2 - sl\}$$

минимальным числом отрезков длины l. Рассмотрим какой-либо из координатных квадратов, для которых эти отрезки являются «верхними» сторонами. Можно было бы за x_{i+1} принять его центр. Однако мы условимся в случае, если «левая» сторона этого квадрата целиком ле-

жит в множестве $K_i(\bar{y}_i+\varepsilon)$, передвинуть его насколько можно «вправо», лишь бы «левая» сторона оставалась в $K_i(\bar{y}_i+\varepsilon)$, а затем, в случае, если «нижняя» сторона лежит в $K_i(\bar{y}_i+\varepsilon)$, передвинуть квадрат насколько можно «вверх», лишь бы «нижняя» сторона оставалась в $K_i(\bar{y}_i+\varepsilon)$. Центр полученного квадрата примем за x_{i+1}.

Чтобы понять, почему описанный алгоритм последовательно-оптимален, достаточно заметить, что задача покрытия построенного нами множества $CK_i(\bar{y}_i+\varepsilon)$ минимальным числом координатных квадратов со стороной l сводится к задачам покрытия полос

$$\{u=(u^1,\ u^2)\mid u\in CK_i(\bar{y}_i+\varepsilon),\ c^2-l\leqslant u^2\leqslant c^2\},$$
$$\{u=(u^1,\ u^2)\mid u\in CK_i(\bar{y}_i+\varepsilon),\ c^2-2l\leqslant u^2\leqslant c^2-l\},\ \ldots$$

минимальным числом таких квадратов. Каждая из этих задач эквивалентна простой задаче покрытия минимальным числом интервалов длины l соответствующих одномерных множеств

$$\{u=(u^1,\ u^2)\mid u\in CK_i(\bar{y}_i+\varepsilon),\ u^2=c^2\},$$
$$\{u=(u^1,\ u^2)\mid u\in CK_i(\bar{y}_i+\varepsilon),\ u^2=c^2-l\},\ \ldots$$

6. О многомерном случае. На основании проведенных в предыдущем пункте построений можно сделать вывод, что даже при $n=2$, когда еще удается конструктивно описать последовательно-оптимальный алгоритм, его использование, как правило, нецелесообразно из-за высокой комбинаторной сложности и непомерно больших требований к объему памяти ЭВМ. Для $n>2$ уже в наиболее простом случае

$$K=\{u\mid 0\leqslant u^i\leqslant 1,\ i=1,\ \ldots,\ n\},$$
$$\rho(u,\ v)=M\max_{i=1,\ldots,n}|u^i-v^i| \tag{4.13}$$

при решении задач оптимального покрытия множеств $CK_i(\bar{y}_i+\varepsilon)$ возникают существенные трудности.

Тем не менее, как отмечалось в п. 2 § 4 гл. 1, знание структуры последовательно-оптимального алгоритма позволяет часто построить эффективные вычислительные методы, являющиеся в том или ином смысле приближениями к последовательно-оптимальному алгоритму и обладающие приемлемой комбинарной сложностью.

Опишем один из возможных подходов. Вначале следует зафиксировать сетку $\{u_1, \ldots, u_N\}$, полный перебор узлов которой должен обеспечить, по оценке вычислителя, решение задачи с нужной точностью (гарантировать обеспечение заданной точности удается не всегда, так как квазиметрика ρ, определяющая функциональный класс, может быть известна неточно и уточняться в процессе решения задачи). В случае (4.13), например, целесообразно зафиксировать кубическую сетку

$$\left\{\left(\frac{j_1}{2m}, \ldots, \frac{j_n}{2m}\right) \middle| j_k = 1, 3, \ldots, 2m-1;\ k = 1, \ldots, n\right\}$$

с $N = m^n$ узлами. В случае неточно известной квазиметрики ρ может оказаться целесообразным использование в качестве точек $u_1, \ldots, u_N$ точки ЛП$_\tau$-последовательности [231, 233, 234, 435], при этом число точек N можно заранее не фиксировать.

Дальнейшее решение задачи состоит в переборе узлов сетки $\{u_1, \ldots, u_N\}$. Если число ее узлов N сравнительно невелико и возможен полный перебор, то никаких проблем не возникает (описанный алгоритм при известном вычислителю ρ есть приближение к оптимальному алгоритму). В противном случае можно использовать следующий метод, основанный на тех же идеях, что и последовательно-оптимальный алгоритм.

Первый шаг определяется соотношением

$$x_1 \in \{u_1, \ldots, u_N\}. \tag{4.14}$$

Пусть сделано i шагов, $i \geqslant 1$. Тогда

$$x_{i+1} \in \{u_j | 1 \leqslant j \leqslant N,\ \varphi_{2i}(u_j) \geqslant \bar{y}_i\} = \{u_1, \ldots, u_N\} \cap CK_i(\bar{y}_i). \tag{4.15}$$

Иными словами, из сетки $\{u_1, \ldots, u_N\}$ следует выбросить точки, в которых заведомо не достигается максимальное значение функции (предполагается, что квазиметрика ρ известна точно, например известно, что ρ имеет вид (4.13) и константа M оценена верно), и в качестве x_{i+1} выбрать одну из оставшихся точек. Конкретные способы выбора точек из множеств (4.14), (4.15) могут быть, как и для последовательно-оптимального алгоритма, разными так же, как и способы программной реализации алгоритма (4.14), (4.15). Можно предложить различные способы программной реализации, основанные на различных методах организации структуры данных (под

структурой данных мы понимаем списки точек u_j, значений функции, точек, в которых значения функции уже вычислены, «выброшенных» точек и т. д.).

Описанный алгоритм можно трактовать как способ выбора начальных точек для методов локальной оптимизации. При этом алгоритм следует дополнить блоком, содержащим один или несколько локальных методов поиска максимума. Данный алгоритм, как и описанные в пп. 1—3 алгоритмы одномерной оптимизации, можно усовершенствовать для решения реальных задач и с помощью иных эвристических приемов. Такими приемами являются использование адаптивной оценки константы Липшица (п. 2 § 4 гл. 2), чередование различных стратегий глобального и локального поиска, применение более реалистических условий остановки (см. по этому поводу обсуждение в п. 3 соотношения между требуемой и фактической точностью решения тестовых задач) и др.

§ 5. Учет погрешности вычислений

В данном параграфе рассматривается вопрос об учете погрешности информационных вычислений, т. е. вычислений значений функции f. При этом, как и ранее, предполагается, что $f \in F = F_\rho$, K — компакт и квазиметрика ρ непрерывна на множестве $K \times K$.

Часто ошибки вычислений считают случайными величинами с известными (обычно нормальными) или не полностью известными законами распределения. К числу таких подходов относится, например, метод стохастической аппроксимации.

Мы будем рассматривать иной подход, предполагая, как и в § 5 гл. 2, что ошибки являются не случайными, а неопределенными величинами, т. е. могут принимать любые значения из некоторого множества. В отличие от § 5 гл. 2, оптимизация по величинам $\delta_1, \ldots, \delta_N$ (максимальным погрешностям информационных вычислений в точках $x_1, \ldots, x_N$) проводиться не будет. Предполагается, что все они не превосходят δ, где δ — известное вычислителю число. Задача состоит в отыскании последовательно-оптимального (а не оптимального пассивного, как в § 5 гл. 2) алгоритма.

1. Постановка задачи. Перейдем к строгим формулировкам. Предположим, что число вычислений N фикси-

ровано и каждое вычисление в произвольной точке $x \in K$ не определяет точно значения $y = f(x)$, а определяет два таких числа $y^- = f^-(x)$, $y^+ = f^+(x)$, что

$$y^- \leqslant y \leqslant y^+, \quad y^+ - y^- \leqslant 2\delta. \tag{5.1}$$

Данная постановка естественно возникает в различных ситуациях, в частности, когда вычислитель находит каждое значение функции f в результате численного решения с заданной точностью δ некоторой задачи, например, $f(x) = \min\limits_{u \in U} g(x, u)$, $f(x) = \int\limits_U g(x, u)\, du$ и т. п. Гарантированная апостериорная погрешность при решении задачи вычисления значения $f(x)$ может оказаться меньше δ, что и отражено в предположениях (5.1).

После окончания информационных вычислений вычислителю известны векторы $x^N = (x_1, \ldots, x_N)$, $y_-^N \overset{\text{def}}{=} (y_1^-, \ldots, y_N^-)$, $y_+^N \overset{\text{def}}{=} (y_1^+, \ldots, y_N^+)$. Положим $z_-^N = (x^N, y_-^N)$, $z_+^N = (x^N, y_+^N)$,

$$F(x^N, y_-^N, y_+^N) = \{f \in F \mid y_j^- \leqslant f(x_j) \leqslant y_j^+,\ j = 1, \ldots, N\},$$

$$\varphi_1(x; z_-^N) = \varphi_1(x; x^N, y_-^N) = \max_{j=1,\ldots,N} \{y_j^- - \rho(x, x_j)\},$$

$$\varphi_2(x; z_+^N) = \varphi_2(x; x^N, y_+^N) = \min_{j=1,\ldots,N} \{y_j^+ + \rho(x, x_j)\}$$

(необходимость ввести в явном виде z_-^N, z_+^N в число аргументов соответственно функций φ_1, φ_2 станет ясна из дальнейших построений). Нетрудно убедиться, что

$$F(x^N, y_-^N, y_+^N) = \{f \in F \mid \varphi_1(x; x^N, y_-^N) \leqslant f(x) \leqslant \varphi_2(x; x^N, y_+^N)\}, \tag{5.2}$$

т. е. φ_1 — точная миноранта, а φ_2 — точная мажоранта для функций из класса $F(x^N, y_-^N, y_+^N)$; однако не обязательно $\varphi_1(x_j; x^N, y_-^N) = y_j^-$, $\varphi_2(x_j; x^N, y_+^N) = y_j^+$, $j = 1, \ldots, N$ (ср. с леммами 1.1 и 1.2 гл. 2).

Для рассматриваемой задачи нам придется заново (хотя и в полном соответствии с ранее введенными понятиями) определить критерий оценки эффективности алгоритмов и понятие последовательно-оптимального алгоритма, так как прежние определения сформулированы в предположении, что значения функции f вычисляются безошибочно.

Руководствуясь стремлением к гарантированным результатам, за приближение к искомой величине $\max\limits_{x\in K} f(x)$ следует принять наименьшее возможное для функций из $F(x^N, y_-^N, y_+^N)$ значение максимума, т. е. значение

$$\max_{x\in K} \varphi_1(x;\ z_-^N) = \max_{i=1,\ldots,N} y_i^- \stackrel{\text{def}}{=} \bar{y}_i.$$

При этом $f(x_{i_0}) \geqslant \varphi_1(x_{i_0};\ z_-^N) = \bar{y}_i$, где $i_0 \in \{1, \ldots, N\}$ определяется из условия $y_{i_0}^- = \bar{y}_i$.

За критерий, как и в случае точных вычислений, примем погрешность определения максимального значения, равную

$$\max_{x\in K} f(x) - \max_{i=1,\ldots,N} f^-(x_i) = \max_{x\in K} f(x) - \bar{y}_N. \qquad (5.3)$$

После N вычислений с результатами x^N, y_-^N, y_+^N максимальная погрешность равна

$$\max_{f\in F(x^N, y_-^N, y_+^N)} \left[\max_{x\in K} f(x) - \bar{y}_N\right] = \max_{x\in K} \varphi_2(x;\ z_+^N) - \bar{y}_N \qquad (5.4)$$

(равенство имеет место в силу (5.2)).

Ситуацию (x, y_-^N, y_+^N) назовем *реализуемой*, если

$$0 \leqslant y_j^+ - y_j^- \leqslant 2\delta, \quad j = 1, \ldots, N, \quad F(x^N, y_-^N, y_+^N) \neq \varnothing.$$

Последовательно-оптимальный по точности алгоритм должен, в соответствии с § 6 гл. 1, выбором точки x_{i+1} очередного информационного вычисления в любой ситуации, сложившейся в ходе решения задачи, минимизировать, в расчете на наихудшее в дальнейшем «поведение» функции f, величину погрешности определения максимального значения (погрешность задается критерием (5.3)).

Точное определение аналогично определению 2 из п. 3 § 6 гл. 1. Именно, назовем в рамках рассматриваемой постановки алгоритм *последовательно-оптимальным по точности*, если точка $(i+1)$-го информационного вычисления x_{i+1}, $i = 0, 1, \ldots, N-1$, выбирается этим алгоритмом из условия реализации внешнего минимума в выражении

$$\min_{x_{i+1}} \max_{y_{i+1}^-, y_{i+1}^+} \ldots \min_{x_N} \max_{y_N^-, y_N^+} \left[\max_{x\in K} \varphi_2(x;\ z_+^N) - \bar{y}_N\right], \qquad (5.5)$$

где x_j, y_j^-, y_j^+, $j = i+1, \ldots, N$, таковы, что ситуация (x^N, y_-^N, y_+^N) реализуема.

2. Последовательно-оптимальный по точности алгоритм. Покажем, что, как и в случае вычислений без погрешности, для построения последовательно-оптимального алгоритма необходимо решить совокупность задач оптимального покрытия. Положим

$$R_0^*(x_1, \ldots, x_N) = \max_{x \in K} \min_{j=1,\ldots,N} \rho(x, x_j),$$

$$R_i^*(x_{i+1}, \ldots, x_N) = \\ = \max_{x \in K} \min \left\{ \min_{j=1,\ldots,i} \left[\rho(x, x_j) - \bar{y}_i + y_j^+ - 2\delta\right], \right. \\ \left. \min_{j=i+1,\ldots,N} \rho(x, x_j) \right\}, \quad i \geqslant 1.$$

Структура последовательно-оптимального алгоритма становится ясной из следующей теоремы (в ее формулировке и доказательстве ситуация (x^N, y_-^N, y_+^N) предполагается реализуемой).

Теорема 5.1 [242]. *При* $i = 0, 1, \ldots, N-1$ *справедливо равенство*

$$\min_{x_{i+1}} \max_{y_{i+1}^-, y_{i+1}^+} \ldots \min_{x_N} \max_{y_N^-, y_N^+} \left[\max_{x \in K} \varphi_2(x; z_+^N) - \bar{y}_N\right] = \\ = \min_{x_{i+1},\ldots,x_N \in K} R_i^*(x_{i+1}, \ldots, x_N) + 2\delta.$$

Доказательство. Преобразуем выражение (5.5). Имеем

$$\max_{y_N^-, y_N^+} \left[\max_{x \in K} \varphi_2(x; z_+^N) - \bar{y}_N\right] = \\ = \max_{y_N^-} \left[\max_{x \in K} \varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta) - \bar{y}_N\right], \quad (5.6)$$

причем требование реализуемости ситуации (x^N, y_-^N, y_+^N) накладывает на y_N^- ограничения

$$\varphi_1(x_N; z_-^{N-1}) - 2\delta \leqslant y_N^- \leqslant \varphi_2(x_N; z_+^{N-1}). \quad (5.7)$$

Найдем сначала величину (5.6), не учитывая ограничений (5.7). Максимум в правой части (5.6) достигается при $y_N^- = \bar{y}_{N-1}$. Действительно, максимизируемая

функция, очевидно, не убывает по y_N^- при $y_N^- \leqslant \bar{y}_{N-1}$. Если же $y_N^- = \bar{y}_{N-1} + \alpha$, $\alpha \geqslant 0$, то

$$\max_{x \in K} \varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta) - \bar{y}_N =$$
$$= \max_{x \in K} \varphi_2(x; x^N, y_+^{N-1}, \bar{y}_{N-1} + \alpha + 2\delta) -$$
$$- \max\{y_1^-, \ldots, y_{N-1}^-, \bar{y}_{N-1} + \alpha\} =$$
$$= \max_{x \in K} \min\{\varphi_2(x; z_+^{N-1}), \bar{y}_{N-1} + \alpha + 2\delta + \rho(x, x_N)\} -$$
$$- \bar{y}_{N-1} - \alpha \leqslant \max_{x \in K} \min\{\varphi_2(x; z_+^{N-1}), \bar{y}_{N-1} + 2\delta +$$
$$+ \rho(x, x_N)\} - \bar{y}_{N-1} = \max_{x \in K} \varphi_2(x; x^N, y_+^{N-1}, \bar{y}_{N-1} + 2\delta) - \bar{y}_{N-1},$$

т. е. значение функции под знаком максимума по y_N^- в правой части (5.6) при $y_N^- \geqslant \bar{y}_{N-1}$ не превосходит значения этой функции при $y_N^- = \bar{y}_{N-1}$.

Покажем теперь, что учет ограничений (5.7) не меняет найденное максимальное значение. Левое неравенство в (5.7) при $y_N^- = \bar{y}_{N-1}$, очевидно, выполняется. Нарушение же правого при $y_N^- = \bar{y}_{N-1}$ (т. е. выполнение неравенства $\bar{y}_{N-1} > \varphi_2(x_N; z_+^{N-1})$) не мешает значению максимума в (5.6) при условии (5.7) совпадать со значением, которое принимает функция под знаком максимума по y_N^- при $y_N^- = \bar{y}_{N-1}$. Чтобы убедиться в этом, достаточно показать, что при любом $x \in K$ функция $\varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta)$ (а стало быть, и функция $\max\limits_{x \in K} \varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta)$) постоянна по y_N^- при $y_N^- \geqslant \varphi_2(x_N; z_+^{N-1}) - 2\delta$.

Для доказательства перепишем последнее неравенство в виде

$$y_N^- \geqslant \min_{j=1,\ldots,N-1} \{y_j^+ + \rho(x_N, x_j)\} - 2\delta = y_{j_0}^+ + \rho(x_N, x_{j_0}) - 2\delta,$$

где $1 \leqslant j_0 \leqslant N-1$. Отсюда

$$y_N^- + 2\delta + \rho(x, x_N) \geqslant y_{j_0}^+ + \rho(x_N, x_{j_0}) + \rho(x, x_N) \geqslant$$
$$\geqslant y_{j_0}^+ + \rho(x, x_{j_0}) \geqslant \min_{j=1,\ldots,N-1} \{y_j^+ + \rho(x, x_j)\} = \varphi_2(x; z_+^{N-1}).$$

Итак, при $y_N^- \geqslant \varphi_2(x_N; z_+^{N-1}) - 2\delta$ имеем

$$\varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta) = \min\{\varphi_2(x; z_+^{N-1}), y_N^- + \\ + 2\delta + \rho(x, x_N)\} = \varphi_2(x; z_+^{N-1}),$$

т. е. действительно функция $\varphi_2(x; x^N, y_+^{N-1}, y_N^- + 2\delta)$ постоянна по y_N^-.

Точно так же показывается, что значение максимума по y_{N-1}^-, y_{N-1}^+ в (5.5), вычисленное с учетом соответствующих ограничений, совпадает со значением максимизируемой функции при $y_{N-1}^- = \bar{y}_{N-2}$, $y_{N-1}^+ = \bar{y}_{N-2} + 2\delta$ и т. д. Тем самым доказана справедливость первого равенства из следующей цепочки (последующие — очевидны):

$$\min_{x_{i+1}} \max_{y_{i+1}^-, y_{i+1}^+} \ldots \min_{x_N} \max_{y_N^-, y_N^+} \left[\max_{x \in K} \varphi_2(x; z_+^N) - \bar{y}_N\right] = \\ = \min_{x_{i+1}, \ldots, x_N} \max_{x \in K} \varphi_2(x; x^N, y_+^i, \bar{y}_i + 2\delta, \ldots, \bar{y}_i + 2\delta) - \bar{y}_i = \\ = \min_{x_{i+1}, \ldots, x_N} \max_{x \in K} \min\{y_1^+ + \rho(x, x_1), \ldots, y_i^+ + \rho(x, x_i), \\ \bar{y}_i + 2\delta + \rho(x, x_{i+1}), \ldots, \bar{y}_i + 2\delta + \rho(x, x_N)\} - \bar{y}_i = \\ = \min_{x_{i+1}, \ldots, x_N} R_i^*(x_{i+1}, \ldots, x_N) + 2\delta. \quad (5.8)$$

В равенствах (5.8) предполагается, что $i \geqslant 1$. Однако, используя (5.8) при $i = 1$, легко получаем утверждение теоремы и в случае $i = 0$. △

Заметим, что (см. (3.4))

$$R_i^*(x_{i+1}, \ldots, x_N) = \\ = \min\left\{r \,\middle|\, \bigcup_{j=1}^{i} S(x_j, \bar{y}_i - y_j^+ + 2\delta + r) \cup \left[\bigcup_{j=i+1}^{N} S(x_j, r)\right] \supset K\right\}.$$

Таким образом, теорема 5.1 сводит задачу построения последовательно-оптимального алгоритма и в случае, когда учитываются погрешности информационных вычислений, к совокупности специального вида задач на оптимальные покрытия. Сравнив выражения для $R_i^*(x_{i+1}, \ldots, x_N)$ и $R_i(x_{i+1}, \ldots, x_N)$ (см. (3.4)), видим, что эти задачи в принципе не отличаются от соответствующих задач на оптимальные покрытия в случае точных вычислений. В частности, если $n = 1$, $\rho(u, v) =$

$= M|u - v|$, то, внося очевидные изменения в алгоритм из п. 1 § 4, получим последовательно-оптимальный по точности алгоритм для рассматриваемого случая приближенных вычислений.

Отметим еще, что если после i информационных вычислений в ситуации x^i, y_-^i, y_+^i гарантированная величина погрешности (максимальная возможная ошибка) не превосходит 2δ, то оставшимися вычислениями гарантировать уменьшение этой погрешности невозможно. Действительно, пусть в ситуации x^i, y_-^i, y_+^i максимальная возможная ошибка не превосходит 2δ. Это означает, что

$$\max_{x\in K} \varphi_2(x; z^i) - \bar{y}_i = \max_{x\in K} \min_{j=1,\ldots,i} [y_j^+ + \rho(x, x_j)] - \bar{y}_i \leqslant 2\delta,$$

откуда

$$\max_{x\in K} \min_{j=1,\ldots,i} [\rho(x, x_j) - \bar{y}_i + y_j^+ - 2\delta] \leqslant 0.$$

Тогда для произвольных $x_{i+1}, \ldots, x_N \in K$

$$R_i^*(x_{i+1}, \ldots, x_N) = \max_{x\in K} \min_{j=1,\ldots,i} [\rho(x, x_j) - \bar{y}_i + y_j^+ - 2\delta],$$

и минимальная погрешность $\min\limits_{x_{i+1},\ldots,x_N} R_i^*(x_{i+1}, \ldots, x_N) + 2\delta$, которую можно гарантировать в ситуации x^i, y_-^i, y_+^i оставшимися $N-i$ информационными вычислениями, совпадает с погрешностью, гарантированной первыми i информационными вычислениями.

§ 6. Оптимальный на один шаг стохастический алгоритм

Пусть $K = [0, 1]$, $\rho(u, v) = M|u - v|$, $F = F_\rho$, т. е.

$$F = \{f \mid |f(u) - f(v)| \leqslant M|u - v|,\ u, v \in [0, 1]\}. \qquad (6.1)$$

Целью данного параграфа является построение оптимального на один шаг стохастического алгоритма в смысле определения (7.15), (7.16) гл. 1 в предположении, что информационными вычислениями являются вычисления точных значений функции f в точках отрезка K, т. е. $X_i = K$, $\Sigma_i = \Sigma$, $i \geqslant 1$, где Σ — множество всех вероятностных мер на σ-алгебре борелевских подмножеств отрезка K.

1. Сведéние задачи построения оптимального на один шаг стохастического алгоритма к решению совокупности антагонистических игр на единичном квадрате. Сначала напомним некоторые из ранее введенных и введем ряд новых обозначений. Именно, для $i \geqslant 1$ положим [1])

$$\varepsilon(x^i, f) = \max_{x \in K} f(x) - \max_{j=1,\ldots,i} f(x_j),$$

$$\bar{y}_i = \max_{j=1,\ldots,i} y_j, \quad \varphi_{2i}(x) = \min_{j=1,\ldots,i} \{y_j + M|x - x_j|\},$$

$$h_a(x) = -M|x-a|, \quad h_a^{i+1}(x) = \max\{\varphi_{2i}(a) - M|x-a|, \bar{y}_i\},$$

$$g_1(a, x) = h_a(a) - h_a(x) = M|x-a|,$$

$$g_{i+1}(a, x) = \varepsilon(x^i, x, h_a^{i+1}) = \max_{\bar{x} \in K} h_a^{i+1}(\bar{x}) -$$

$$- \max\{h_a^{i+1}(x_1), \ldots, h_a^{i+1}(x_i), h_a^{i+1}(x)\} = h_a^{i+1}(a) - h_a^{i+1}(x).$$

Лемма 6.1. *Пусть функциональный класс F определен формулой* (6.1), σ_j — *оптимальная смешанная стратегия минимизирующего игрока в игре на единичном квадрате с функцией выигрыша g_j. Тогда совокупность вероятностных мер $\{\sigma_j\}_{j \geqslant 1}$ определяет алгоритм, оптимальный на один шаг на классе F, среди стохастических последовательных алгоритмов, т. е. σ_1 удовлетворяет равенству* (7.5) *гл.* 1, *а σ_{i+1}, $i \geqslant 1$,— равенству* (7.16) *гл.* 1.

Доказательство. Пусть $f \in F(z^i)$, $\max_{x \in K} f(x) = f(a)$, $i \geqslant 1$. Для произвольного $x \in K$, очевидно, $f(x) \geqslant f(a) - M|x-a|$, откуда

$$\begin{aligned}\varepsilon(x^i, x, f) &= f(a) - \max\{f(x), \bar{y}_i\} \leqslant \\ &\leqslant f(a) - \max\{f(a) - M|x-a|, \bar{y}_i\} \leqslant \\ &\leqslant \varphi_{2i}(a) - \max\{\varphi_{2i}(a) - M|x-a|, \bar{y}_i\} = h_a^{i+1}(a) - \\ &- h_a^{i+1}(x) = \varepsilon(x^i, x, h_a^{i+1}) = g_{i+1}(a, x). \qquad (6.2)\end{aligned}$$

Заметим, что для функции $k_a^{i+1}(x) \overset{\text{def}}{=} \min\{h_a^{i+1}(x), \varphi_{2i}(x)\}$ справедливо равенство $\varepsilon(x^i, x, h_a^{i+1}) = \varepsilon(x^i, x, k_a^{i+1})$ и $k_a^{i+1} \in F(z^i)$. В силу неравенства (6.2) и произвольности $x \in K$

---

[1]) Для упрощения обозначений не указывается в явном виде, как и для ранее введенной функции φ_{2i}, зависимость h_a^{i+1} и g_{i+1} от z^i.

для произвольной меры $\sigma \in \Sigma$ имеем

$$\int_K \varepsilon(x^i, x, f)\,\sigma\{dx\} \leqslant \int_K \varepsilon(x^i, x, h_a^{i+1})\,\sigma\{dx\} =$$
$$= \int_K \varepsilon(x^i, x, k_a^{i+1})\,\sigma\{dx\} \leqslant \max_{a\in K} \int_K \varepsilon(x^i, x, k_a^{i+1})\,\sigma\{dx\},$$

а в силу произвольности $f \in F(z^i)$ —

$$\max_{f\in F(z^i)} \int_K \varepsilon(x^i, x, f)\,\sigma\{dx\} \leqslant \max_{a\in K} \int_K \varepsilon(x^i, x, k_a^i)\,\sigma\{dx\}.$$

(Очевидно, что для функционального класса (6.1) все максимальные значения, фигурирующие здесь и ниже, достигаются.) Далее, поскольку $k_a^{i+1} \in F(z^i)$ при всех $a \in K$, то справедливо и противоположное неравенство. Используя равенство $\varepsilon(x^i, x, k_a^{i+1}) = g_{i+1}(a, x)$, получаем отсюда

$$\max_{f\in F(z^i)} \int_K \varepsilon(x^i, x, f)\,\sigma\{dx\} = \max_{a\in K} \int_K g_{i+1}(a, x)\,\sigma\{dx\}. \quad (6.3)$$

Пусть σ_{i+1} — оптимальная смешанная стратегия минимизирующего игрока в игре на единичном квадрате с функцией выигрыша g_{i+1}. Это означает (см., например, [122]), что

$$\max_{a\in K} \int_K g_{i+1}(a, x)\,\sigma_{i+1}\{dx\} = \min_{\sigma\in\Sigma} \max_{a\in K} \int_K g_{i+1}(a, x)\,\sigma\{dx\}.$$

Используя то, что равенство (6.3) справедливо при всех $\sigma \in \Sigma$, получаем отсюда равенство (7.16) гл. 1. Справедливость равенства (7.15) гл. 1 следует из леммы 2.1. △

2. Решение антагонистических игр. Решение игры с функцией выигрыша g_1 не представляет труда. Обозначим через I_c вероятностную меру, сосредоточенную в точке c.

Лемма 6.2. *Стратегии* $\sigma_1 = I_{1/2}$, $\tau_1 = I_0/2 + I_1/2$ — *оптимальные смешанные стратегии соответственно минимизирующего и максимизирующего игроков, а величина* $M/2$ — *значение игры на единичном квадрате с функцией выигрыша* g_1. △

Доказательство состоит в тривиальной проверке, которую мы опускаем. Заметим, что игра с функцией выигрыша g_1 является частным случаем игры, рассмотренной в доказательстве теоремы 2.2.

Обратимся к решению игры на единичном квадрате с функцией выигрыша g_{i+1} при $i \geqslant 1$. Пусть множество $\{x \mid \varphi_{2i}(x) > \bar{y}_i\}$ состоит из n непересекающихся промежутков, $n \leqslant i+1$. Обозначим через b_j, d_j концы j-го промежутка. Тогда

$$P_i \stackrel{\text{def}}{=} \{x \mid \varphi_{2i}(x) \geqslant \bar{y}_i\} = \bigcup_{j=1}^{n} [b_j, d_j].$$

Положим $\alpha_j = b_j - d_j$, если $b_j \neq 0, d_j \neq 1$, и $\alpha_j = 2(d_j - b_j)$,

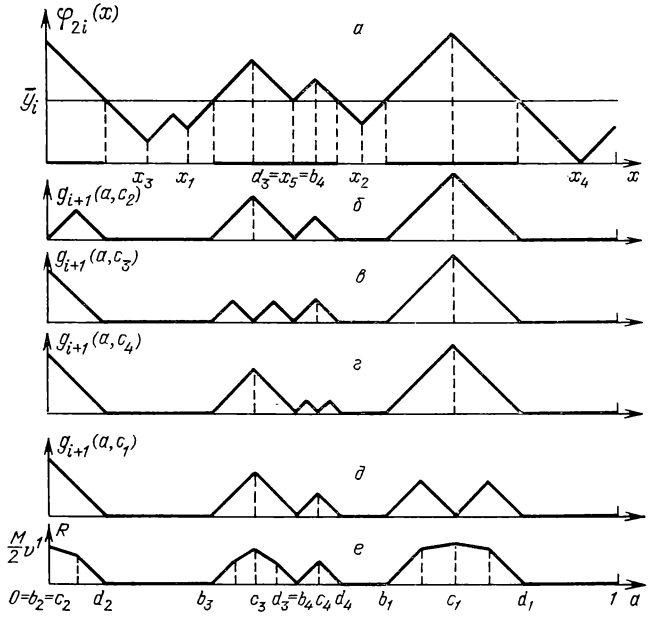

Рис. 10. К вопросу об оптимальности стратегии σ_{i+1} минимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае $q_1^1 \leqslant 1/2$

если $b_j = 0$ или $d_j = 1$. Будем считать, что промежутки занумерованы таким образом, что

$$\alpha_1 \geqslant \alpha_2 \geqslant \ldots \geqslant \alpha_n > 0. \tag{6.4}$$

Положим, далее, $c_j = \arg \max\limits_{x \in [b_j, d_j]} \varphi_{2i}(x)$. Ясно, что $c_j = 0$, если $b_j = 0$; $c_j = 1$, если $d_j = 1$; $c_j = (b_j + d_j)/2$ в остальных случаях. На рис. 10, *а* изображен случай $i = 5$, $n = 4$.

Решение игры можно получить при помощи рассуждений по следующей схеме. Сначала из соображений доминирования получается, что точки спектров оптимальных стратегий для каждого из игроков достаточно искать в множестве $P(z^i)$. Далее показывается, что эти спектры содержат на каждом из отрезков $[b_j, d_j]$, $j > 1$, не более одной точки, а именно c_j. На отрезке $[b_1, d_1]$ в спектры оптимальных стратегий игроков, кроме точки c_1, могут входить также точки $\xi_1 = b_1 + (d_1 - b_1)/3$, $\eta_1 = b_1 + 2(d_1 - b_1)/3$, если $b_1 \neq 0$, $d_1 \neq 1$. Если же $b_1 = 0$ или $d_1 = 1$, то, кроме c_1, в спектры оптимальных стратегий может входить точка $\zeta_1 = (b_1 + d_1)/2$. Таким образом, спектры оптимальных стратегий содержат не более $n + 2$ точек каждый. Это позволяет свести решение игры на единичном квадрате к решению матричных игр.

Доказательство утверждения о точках спектров оптимальных стратегий, содержащихся в отрезках $[b_j, d_j]$, $j > 1$, не представляет особого труда. Для отыскания дополнительных точек спектров на $[b_1, d_1]$ необходимо решить игру с функцией выигрыша g_{i+1} на квадрате $[b_1, d_1] \times [b_1, d_1]$. На этом квадрате функция g_{i+1} при $b_1 \neq 0$, $d_1 \neq 1$ является бабочкообразной функцией выигрыша [122]. Смешанная стратегия $I_{\xi_1}/2 + I_{\eta_1}/2$ оптимальна для каждого из игроков. Мы не станем приводить доказательства этих и других утверждений, необходимых при естественном ходе отыскания решения игры. Взамен дадим более короткую формальную схему со всеми необходимыми доказательствами.

Рассмотрим игру с матрицей $A^1 = (a^1_{jk}) \overset{\text{def}}{=} (g_{i+1}(c_j, c_k))$, $j, k = 1, \ldots, n$. Используя формулу $g_{i+1}(a, x) = h_a^{i+1}(a) - h_a^{i+1}(x)$, получаем

$$A^1 = \frac{M}{2} \begin{pmatrix} 0 & \alpha_1 & \alpha_1 & \ldots & \alpha_1 \\ \alpha_2 & 0 & \alpha_2 & \ldots & \alpha_2 \\ \alpha_3 & \alpha_3 & 0 & \ldots & \alpha_3 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \alpha_n & \alpha_n & \alpha_n & \ldots & 0 \end{pmatrix}^{1)} .$$

Для решения игры с матрицей A^1 при сделанном ранее предположении (6.4) положим $v_j^1 = (j-1)/(\alpha_1^{-1} + \alpha_2^{-1} + \ldots + \ldots + \alpha_j^{-1})$, $j = 1, \ldots, n$.

[1]) Для упрощения обозначений не указывается в явном виде зависимость матрицы A^1, определяемых ниже матриц A^2, A^3, а также элементов этих трех матриц ни от z^i, ни от номера шага.

Лемма 6.3. *Пусть* $v_1^1 < v_2^1 < \ldots < v_{j_1}^1$, *а* $v_{j_1}^1 \geqslant v_{j_1+1}^1$, *или* $j_1 = n$. *Тогда* $Mv^1/2 \overset{\text{def}}{=} Mv_{j_1}^1/2$ — *значение игры с матрицей* A^1,

$$(q_1^1, q_2^1, \ldots, q_{j_1}^1, q_{j_1+1}^1, \ldots, q_n^1) = \\ = (1 - v^1/\alpha_1, 1 - v^1/\alpha_2, \ldots, 1 - v^1/\alpha_{j_1}, 0, \ldots, 0)$$

— *оптимальная стратегия минимизирующего игрока, а*

$$(p_1^1, p_2^1, \ldots, p_{j_1}^1, p_{j_1+1}^1, \ldots, p_n^1) = \\ = \left(\frac{1}{\alpha_1}\cdot\frac{v^1}{j_1-1}, \frac{1}{\alpha_2}\cdot\frac{v^1}{j_1-1}, \ldots, \frac{1}{\alpha_{j_1}}\cdot\frac{v^1}{j_1-1}, 0, \ldots, 0\right)$$

— *оптимальная стратегия максимизирующего игрока.*

Доказательство. Для обозначения эквивалентных неравенств будем пользоваться символом $\Leftrightarrow$. Непосредственно проверяется, что

$$v_j^1 < v_{j+1}^1 \Leftrightarrow v_j^1 < \alpha_{j+1} \Leftrightarrow v_{j+1}^1 < \alpha_{j+1}, \quad j = 1, \ldots, n-1, \tag{6.5}$$

$$\sum_{j=1}^{n} p_j^1 = 1, \quad p_j^1 \geqslant 0, \quad j = 1, \ldots, n, \quad \sum_{k=1}^{n} q_k^1 = 1, \tag{6.6}$$

$$\sum_{j=1}^{n} a_{jk}^1 p_j^1 = \frac{M}{2} v^1, \quad k \leqslant j_1, \tag{6.7}$$

$$\sum_{j=1}^{n} a_{jk}^1 p_j^1 = \frac{j_1}{j_1-1}\cdot\frac{M}{2} v^1 > \frac{M}{2} v^1, \quad k \geqslant j_1 + 1, \tag{6.8}$$

$$\sum_{k=1}^{n} a_{jk}^1 q_k^1 = \frac{M}{2} v^1, \quad j \leqslant j_1. \tag{6.9}$$

Используя (6.4), (6.5) и условие $v_{j_1}^1 \geqslant v_{j_1+1}^1$, получаем

$$\sum_{k=1}^{n} a_{jk}^1 q_k^1 = \frac{M}{2} \alpha_j \leqslant \frac{M}{2} \alpha_{j_1+1} \leqslant \frac{M}{2} v^1, \quad j \geqslant j_1 + 1. \tag{6.10}$$

Выпишем, наконец, цепочку эквивалентных в силу (6.5) неравенств, первое из которых справедливо по условию леммы:

$$v_{j-1}^1 < v_{j_1}^1 \Leftrightarrow v_{j_1-1}^1 < \alpha_{j_1} \Leftrightarrow v^1 < \alpha_{j_1} \Leftrightarrow q_{j_1}^1 > 0.$$

Отсюда в силу (6.4) получаем $q_k^1 > 0$, $k = 1, \ldots, j_1$, что вместе с соотношениями (6.6)—(6.10) доказывает лемму. △

Лемма 6.4. *Если* $q_1^1 \leqslant 1/2$, *то*

$$\sigma_{i+1} = \sum_{i=1}^{n} q_k^1 I_{c_k}, \quad \tau_{i+1} = \sum_{j=1}^{n} p_j^1 I_{c_j}$$

суть оптимальные смешанные стратегии соответственно минимизирующего и максимизирующего игроков, а $Mv^1/2$ — *значение игры на единичном квадрате с функцией выигрыша* g_{i+1}.

Доказательство. Заметим, что $q_k^1 \leqslant 1/2$, $k = 1, \ldots, n$, так как, по условию леммы и в силу (6.4), $1/2 \geqslant q_1^1 \geqslant q_2^1 \geqslant \ldots \geqslant q_n^1$. Поэтому функция

$$R \stackrel{\text{def}}{=} \int_K g_{i+1}(a, x)\, \sigma_{i+1}\{dx\} = \sum_{i=1}^{n} g_{i+1}(a, c_k)\, q_k^1,$$

рассматриваемая как функция аргумента a, достигает максимального на отрезке $[b_j, d_j]$ значения в точке c_j, $j = 1, \ldots, n$. Это ясно из рис. 10, *б—е*, на котором изображены графики функций $g_{i+1}(a, c_j)$, $j = 1, \ldots, n$, а также график функции R (все они рассматриваются как функции аргумента a) в случае, соответствующем рис. 10, *а*. Таким образом,

$$\max_{a \in K} R = \max_{j=1,\ldots,n} \sum_{k=1}^{n} g_{i+1}(c_j, c_k)\, q_k^1 = \\ = \max_{j=1,\ldots,n} \sum_{k=1}^{n} a_{jk}^1 q_k^1 = \frac{M}{2} v^1. \quad (6.11)$$

Рассмотрим теперь

$$R' \stackrel{\text{def}}{=} \int_K g_{i+1}(a, x)\, \tau_{i+1}\{da\} = \sum_{j=1}^{n} g_{i+1}(c_j, x)\, p_j^1$$

как функцию аргумента x. Очевидно (см. рис. 11), что эта функция достигает минимального на отрезке $[b_k, d_k]$

значения в точке c_k, $k=1, \ldots, n$. Отсюда

$$\min_{x\in K} R' = \min_{k=1,\ldots,n} \sum_{j=1}^{n} g_{i+1}(c_j, c_k)\, p_j^1 =$$

$$= \min_{k=1,\ldots,n} \sum_{j=1}^{n} a_{jk}^1 p_j^1 = \frac{M}{2} v^1. \quad (6.12)$$

Равенства (6.11), (6.12) доказывают лемму. △

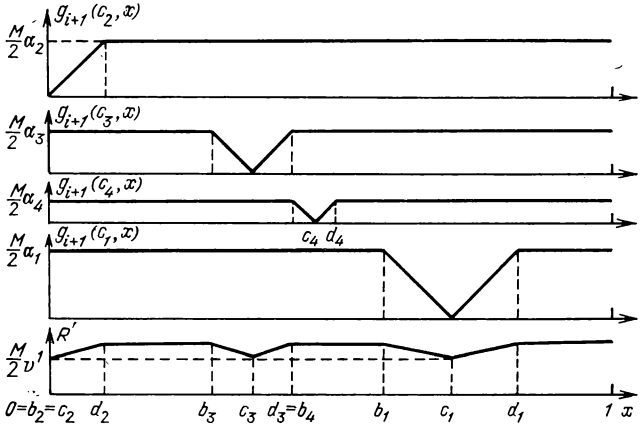

Рис. 11. К вопросу об оптимальности стратегии τ_{i+1} максимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае $q_1^1 \leqslant 1/2$

Рассмотрим теперь игру с матрицей

$$A^2 = (a_{jk}^2)_{j,k=1,\ldots,n} \stackrel{\text{def}}{=} \frac{M}{2} \begin{pmatrix} \alpha_1/3 & 2\alpha_1/3 & 2\alpha_1/3 & \ldots & 2\alpha_1/3 \\ \alpha_2 & 0 & \alpha_2 & \ldots & \alpha_2 \\ \alpha_3 & \alpha_3 & 0 & \ldots & \alpha_3 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \alpha_n & \alpha_n & \alpha_n & \ldots & 0 \end{pmatrix}.$$

Положим $v_j^2 = j/(3\alpha_1^{-1} + \alpha_2^{-1} + \ldots + \alpha_j^{-1})$, $j=1, \ldots, n$.

Лемма 6.5. *Пусть* $v_1^2 < v_2^2 < \ldots < v_{j_2}^2$, *а* $v_{j_2}^2 \geqslant v_{j_2+1}^2$ *или* $j_2 = n$. *Пусть, кроме того,* $q_1^1 > 1/2$. *Тогда* $Mv^2/2 \stackrel{\text{def}}{=} Mv_{j_2}^2/2$ — *значение игры с матрицей* A^2,

$$(q_1^2, q_2^2, \ldots, q_{j_2}^2, q_{j_2+1}^2, \ldots, q_n^2) =$$
$$= (2 - 3v^2/\alpha_1,\ 1 - v^2/\alpha_2, \ldots, 1 - v^2/\alpha_{j_2}, 0, \ldots, 0)$$

есть оптимальная стратегия минимизирующего игрока, при этом $q_1^2 > 1/2$, *а*

$$(p_1^2, p_2^2, \ldots, p_{j_2}^2, p_{j_2+1}^2, \ldots, p_n^2) = \\ = \left(\frac{3}{\alpha_1}\cdot\frac{v^2}{j_2}, \frac{1}{\alpha_2}\cdot\frac{v^2}{j_2}, \ldots, \frac{1}{\alpha_{j_2}}\cdot\frac{v^2}{j_2}, 0, \ldots, 0\right)$$

есть оптимальная стратегия максимизирующего игрока.

Доказательство. Непосредственно проверяется, что

$$v_j^2 < v_{j+1}^2 \Leftrightarrow v_j^2 < \alpha_{j+1} \Leftrightarrow v_{j+1}^2 < \alpha_{j+1}, \quad j = 1, \ldots, n-1, \tag{6.13}$$

$$\sum_{j=1}^{n} p_j^2 = 1, \quad p_j^2 \geqslant 0, \quad j = 1, \ldots, n,$$

$$\sum_{k=1}^{n} q_k^2 = 1, \quad \sum_{j=1}^{n} a_{jk}^2 p_j^2 = \frac{M}{2} v^2, \quad k \leqslant j_2,$$

$$\sum_{j=1}^{n} a_{jk}^2 p_j^2 = \frac{j_2+1}{j_2}\cdot\frac{M}{2} v^2 > \frac{M}{2} v^2, \quad k \geqslant j_2 + 1,$$

$$\sum_{k=1}^{n} a_{jk}^2 q_k^2 = \frac{M}{2} v^2, \quad j \leqslant j_2.$$

Используя (6.4), (6.13) и условие $v_{j_2}^2 \geqslant v_{j_2+1}^2$, получаем

$$\sum_{k=1}^{n} a_{jk}^2 q_k^2 = \frac{M}{2}\alpha_j \leqslant \frac{M}{2}\alpha_{j_2+1} \leqslant \frac{M}{2} v^2, \quad j \geqslant j_2 + 1.$$

Далее, так же как при доказательстве леммы 6.3, убеждаемся, что $q_k^2 > 0$, $k = 2, \ldots, j_2$. Для завершения доказательства остается убедиться, что $q_1^2 > 1/2$.

По условию, $q_1^1 > 1/2$, откуда легко выводим эквивалентное неравенство

$$\alpha_2^{-1} + \ldots + \alpha_{j_1}^{-1} > (2j_1 - 3)\,\alpha_1^{-1}. \tag{6.14}$$

Пусть $j_1 > j_2$. Тогда $v_{j_2}^1 < v_{j_2+1}^1$, $v_{j_2+1}^2 \leqslant v_{j_2}^2$. В силу (6.5) и (6.13) $v_{j_2}^1 < \alpha_{j_2+1} \leqslant v_{j_2}^2$. Имеем $v_{j_2}^1 < v_{j_2}^2 \Leftrightarrow \alpha_2^{-1} + \ldots$ $\ldots + \alpha_{j_2}^{-1} > (2j_2 - 3)\,\alpha_1^{-1} \Leftrightarrow q_1^2 > 1/2$.

Если $j_1 = j_2$, то условие (6.14) дает как раз необходимое неравенство $q_1^2 > 1/2$.

Предположим, наконец, что $j_1 < j_2$. Тогда $v_{j_1}^2 < v_{j_1+1}^2$, $v_{j_1+1}^1 \leqslant v_{j_1}^1$. В силу (6.4) и (6.13) $v_{j_1}^2 < \alpha_{j_1+1} \leqslant v_{j_1}^1$. Имеем $v_{j_1}^2 < v_{j_1}^1 \Leftrightarrow \alpha_2^{-1} + \ldots + \alpha_{j_1}^{-1} < (2j_1 - 3)\alpha_1^{-1}$, что противоречит неравенству (6.14). Неравенство $q_1^2 > 1/2$ и тем самым лемма полностью доказаны. △

Напомним введенные ранее обозначения: $\xi_1 = b_1 + (d_1 - b_1)/3$, $\eta_1 = b_1 + 2(d_1 - b_1)/3$. Кроме того, положим $\bar{q}_1^2 = \sum_{k=2}^{n} q_k^2$.

Лемма 6.6. *Если* $q_1^1 > 1/2$, $b_1 \neq 0$, $d_1 \neq 1$, *то*

$$\sigma_{i+1} = \bar{q}_1^2 I_{c_1} + (q_1^2 - \bar{q}_1^2)\left(\frac{1}{2} I_{\xi_1} + \frac{1}{2} I_{\eta_1}\right) + \sum_{k=2}^{n} q_k^2 I_{c_k},$$

$$\tau_{i+1} = p_1^2\left(\frac{1}{2} I_{\xi_1} + \frac{1}{2} I_{\eta_1}\right) + \sum_{j=2}^{n} p_j^2 I_{c_j}$$

суть оптимальные смешанные стратегии соответственно минимизирующего и максимизирующего игроков, а $Mv^2/2$ — *значение игры на единичном квадрате с функцией выигрыша* g_{i+1}.

Доказательство. Отметим, что при $q_1^1 > 1/2$, по лемме 6.5, $q_1^2 > 1/2$ и, следовательно, $\bar{q}_1^2 < 1/2$ и $q_1^2 - \bar{q}_1^2 > 0$. Таким образом, σ_{i+1} и, очевидно, τ_{i+1} — смешанные стратегии. Имеем

$$R = Q + (q_1^2 - \bar{q}_1^2)\, G,$$

где

$$Q = \bar{q}_1^2 g_{i+1}(a, c_1) + \sum_{k=2}^{n} q_k^2 g_{i+1}(a, c_k),$$
$$G = [g_{i+1}(a, \xi_1) + g_{i+1}(a, \eta_1)]/2.$$

Если $a \notin [b_1, d_1]$, то $g_{i+1}(a, c_1) = g_{i+1}(a, \xi_1) = g_{i+1}(a, \eta_1)$, поэтому $R = \sum_{k=1}^{n} q_k^2 g_{i+1}(a, c_k)$ и максимум по a на отрезке $[b_j, d_j]$, $j \geqslant 2$, достигается в точке c_j (обоснование то же, что в доказательстве леммы 6.4: достаточно заметить, что

$q_j^2 < \sum_{k \neq j} q_k^2$, $j \geqslant 2$). Таким образом,

$$\max_{a \in [b_j, d_j]} R = \int_K g_{i+1}(c_j, x)\, \sigma_{i+1}\{dx\} = \sum_{k=1}^{n} a_{jk}^2 q_k^2, \quad j \geqslant 2. \quad (6.15)$$

Из рис. 12 ясно, что

$$\operatorname{Arg} \max_{a \in [b_1, d_1]} R = [b_1 + (d_1 - b_1)/4,\ b_1 + 3(d_1 - b_1)/4],$$

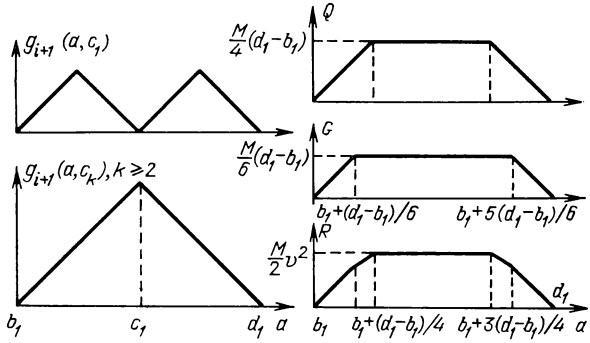

Рис. 12. К вопросу об оптимальности стратегии σ_{i+1} минимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае $q_1^1 > 1/2$, $b_1 \neq 0$, $d_1 \neq 1$

откуда

$$\max_{a \in [b_1, d_1]} R = \int_K g_{i+1}(c_1, x)\, \sigma_{i+1}\{dx\} =$$

$$= \frac{M}{2}\left[\bar{q}_1^2 \cdot 0 + (q_1^2 - \bar{q}_1^2)\frac{\alpha_1}{3} + \sum_{k=2}^{n} q_k^2 \alpha_1\right] =$$

$$= \frac{M}{2}\left(q_1^2 \frac{\alpha_1}{3} + \sum_{k=2}^{n} q_k^2 \frac{2\alpha_1}{3}\right) = \sum_{k=1}^{n} a_{1k}^2 q_k^2. \quad (6.16)$$

Замечая, что $R = 0$ при $a \notin \bigcup_{j=1}^{n} [b_j, d_j]$, имеем, вследствие (6.15), (6.16),

$$\max_{a \in K} R = \max_{j=1,\ldots,n} \sum_{k=1}^{n} a_{jk}^2 q_k^2 = \frac{M}{2} v^2. \quad (6.17)$$

Далее докажем, что функция

$$R' = p_1^2 G' + \sum_{j=2}^{n} p_j^2 g_{i+1}(c_j, x),$$

где $G' = [g_{i+1}(\xi_1, x) + g_{i+1}(\eta_1, x)]/2$, достигает минимума по x на отрезке $[b_k, d_k]$ в точке c_k. Для $k=1$ это ясно из рис. 13:

$$\operatorname{Arg} \min_{x \in [b_1, d_1]} R' = [b_1 + (d_1 - b_1)/3, \; b_1 + 2(d_1 - b_1)/3];$$

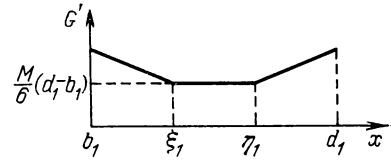

Рис. 13. К вопросу об оптимальности стратегии τ_{i+1} максимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае $q_1^1 > 1/2$, $b_1 \neq 0$, $d_1 \neq 1$

для $k \geqslant 2$ — показывается так же, как при доказательстве леммы 6.4. Отсюда легко получаем, что

$$\min_{x \in K} R' = \min_{k=1,\ldots,n} \sum_{j=1}^{n} a_{jk}^2 p_j^2 = \frac{M}{2} v^2. \tag{6.18}$$

Равенства (6.17), (6.18) доказывают лемму. △

Рассмотрим теперь еще одну (и последнюю) игру — с матрицей

$$A^3 = (a_{jk}^3)_{j,k=1,\ldots,n} \stackrel{\text{def}}{=} \frac{M}{2} \begin{pmatrix} \alpha_1/4 & 3\alpha_1/4 & 3\alpha_1/4 & \ldots & 3\alpha_1/4 \\ \alpha_2 & 0 & \alpha_2 & \ldots & \alpha_2 \\ \alpha_3 & \alpha_3 & 0 & \ldots & \alpha_3 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \alpha_n & \alpha_n & \alpha_n & \ldots & 0 \end{pmatrix}.$$

Положим $v_j^3 = (j - 1/2)(2\alpha_1^{-1} + \alpha_2^{-1} + \ldots + \alpha_j^{-1})$, $j = 1, \ldots, n$.

Лемма 6.7. *Пусть* $v_1^3 < v_2^3 < \ldots < v_{j_3}^3$, *а* $v_{j_3}^3 \geqslant v_{j_3+1}^3$, *или* $j_3 = n$. *Пусть, кроме того,* $q_1^1 > 1/2$. *Тогда* $Mv^3/2 \stackrel{\text{def}}{=} Mv_{j_3}^3/2$ — *значение игры с матрицей* A^3,

$$(q_1^3, q_2^3, \ldots, q_{j_3}^3, q_{j_3+1}^3, \ldots, q_n^3) = (3/2 - 2v^3/\alpha_1, 1 - v^3/\alpha_2, \ldots, 1 - v^3/\alpha_{j_3}, 0, \ldots, 0)$$

есть оптимальная стратегия минимизирующего игрока,

при этом $q_1^3 > 1/2$, *а*

$$(p_1^3, p_2^3, \ldots, p_{j_3}^3, p_{j_3+1}^3, \ldots, p_n^3) = \left(\frac{2}{\alpha_1}\cdot\frac{v^3}{j_3-1/2}, \frac{1}{\alpha_2}\cdot\frac{v^3}{j_3-1/2}, \ldots, \frac{1}{\alpha_{j_3}}\cdot\frac{v^3}{j_3-1/2}, 0, \ldots, 0\right)$$

есть оптимальная стратегия максимизирующего игрока. △

Доказательство леммы аналогично доказательству леммы 6.5.

Напомним введенное ранее для случая $b_1 = 0$ или $d_1 = 1$ обозначение $\zeta_1 = (b_1 + d_1)/2$. Кроме того, положим $\bar{q}_1^3 = \sum_{k=2}^{n} q_k^3$.

Лемма 6.8. *Если* $q_1^1 > 1/2$ *и, кроме того,* $b_1 = 0$ *или* $d_1 = 1$, *то*

$$\sigma_{i+1} = \bar{q}_1^3 I_{c_1} + (q_1^3 - \bar{q}_1^3)\left(\frac{1}{2} I_{\zeta_1} + \frac{1}{2} I_{c_1}\right) + \sum_{k=2}^{n} q_k^3 I_{c_k},$$

$$\tau_{i+1} = p_1^3\left(\frac{1}{2} I_{\zeta_1} + \frac{1}{2} I_{c_1}\right) + \sum_{j=2}^{n} p_j^2 I_{c_j}$$

суть оптимальные стратегии соответственно минимизирующего и максимизирующего игроков, а $Mv^3/2$ — *значение игры на единичном квадрате с функцией выигрыша* g_{i+1}.

Доказательство. Отметим, что при $q_1^1 > 1/2$, по лемме 6.7, $q_1^3 > 1/2$, и, следовательно, $\bar{q}_1^3 < 1/2$ и $q_1^3 - \bar{q}_1^3 > 0$. Таким образом, σ_{i+1} и, очевидно, τ_{i+1} — смешанные стратегии. Имеем

$$R = Q' + (q_1^3 - \bar{q}_1^3) G'',$$

где

$$Q' = \bar{q}_1^3 g_{i+1}(a, c_1) + \sum_{k=2}^{n} q_k^3 g_{i+1}(a, c_k),$$
$$G'' = [g_{i+1}(a, \zeta_1) + g_{i+1}(a, c_1)]/2.$$

Если $a \notin [b_1, d_1]$, то $g_{i+1}(a, c_1) = g_{i+1}(a, \zeta_1)$, а потому $R = \sum_{k=1}^{n} q_k^3 g_{i+1}(a, c_k)$, и максимум по a на отрезке $[b_j, d_j]$, $j \geqslant 2$, достигается в точке c_j. Обоснование то же, что в

доказательстве леммы 6.4: достаточно заметить, что $q_j^3 < \sum_{k \neq j} q_k^3$, $j \geqslant 2$. Таким образом,

$$\max_{a \in [b_j, d_j]} R = \int_K g_{i+1}(c_j, x)\, \sigma_{i+1}\{dx\} = \sum_{k=1}^{n} a_{jk}^3 q_k^3, \quad j \geqslant 2. \quad (6.19)$$

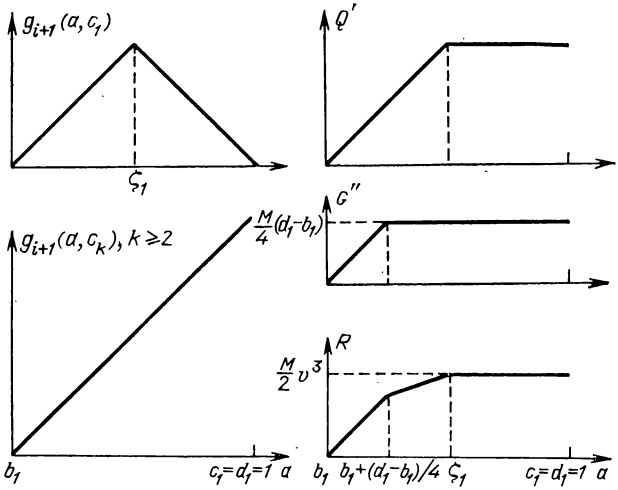

Рис. 14. К вопросу об оптимальности стратегии σ_{i+1} минимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае, когда $q_1^1 > 1/2$ и, кроме того, $b_1 = 0$ или $d_1 = 1$

Из рис. 14 ясно, что $\operatorname{Arg} \max_{a \in [b_1, d_1]} R = [\zeta_1, c_1]$, откуда

$$\max_{a \in [b_1, d_1]} R = \int_K g_{i+1}(c_1, x)\, \sigma_{i+1}\{dx\} =$$

$$= \frac{M}{2}\left[\bar{q}_1^3 \cdot 0 + (q_1^3 - \bar{q}_1^3)\frac{\alpha_1}{4} + \sum_{k=2}^{n} q_k^3 \alpha_1\right] =$$

$$= \frac{M}{2}\left(q_1^3 \frac{\alpha_1}{4} + \sum_{k=2}^{n} q_k^3 \frac{3\alpha_1}{4}\right) = \sum_{k=1}^{n} a_{1k}^3 q_k^3. \quad (6.20)$$

Замечая, что $R = 0$ при $a \notin \bigcup_{j=1}^{n} [b_j, d_j]$, имеем, вследствие (6.19), (6.20),

$$\max_{a \in K} R = \max_{j=1,\ldots,n} \sum_{k=1}^{n} a_{jk}^3 q_k^3 = \frac{M}{2} v^3. \quad (6.21)$$

Далее покажем, что функция

$$R' = p_1' G''' + \sum_{j=2}^{n} p_j^3 g_{i+1}(c_j, x),$$

где $G''' = [g_{i+1}(\zeta_1, x) + g_{i+1}(c_1, x)]/2$, достигает минимума по x на отрезке $[b_k, d_k]$ в точке c_k. Для $k=1$ это ясно из рис. 15: $\operatorname{Arg}\min\limits_{x\in[b_1,d_1]} R' = [\zeta_1, c_1]$; для $k \geqslant 2$ показывается так же, как при доказательстве леммы 6.4. Отсюда легко получаем, что

$$\min_{x\in K} R' = \min_{k=1,\ldots,n} \sum_{j=1}^{n} a_{jk}^3 p_j^3 = \frac{M}{2} v^3. \tag{6.22}$$

Равенства (6.21), (6.22) доказывают лемму. △

3. Оптимизационный алгоритм. Опишем в заключение стохастический алгоритм поиска максимума функции f, удовлетворяющей на отрезке [0, 1] условию Липшица с известной константой M, т. е. принадлежащей функциональному классу (6.1).

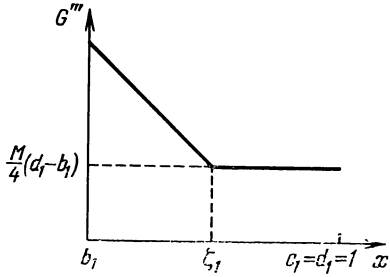

Рис. 15. К вопросу об оптимальности стратегии τ_{i+1} максимизирующего игрока в игре с функцией выигрыша g_{i+1} в случае, когда $q_1^1 > 1/2$ и, кроме того, $b_1 = 0$ или $d_1 = 1$

На первом шаге полагаем $x_1 = 1/2$ и вычисляем $y_1 = f(x_1)$. Пусть сделано i шагов: $y_j = f(x_j)$, $j = 1, \ldots, i$, $i \geqslant 1$.

На $(i+1)$-м шаге производим следующие действия. По формулам п. 2 вычисляем величины n, b_j, d_j, α_j, c_j для $j = 1, \ldots, n$. Затем вычисляем v^1, $q_1^1, \ldots, q_n^1$.

Если $q_1^1 \leqslant 1/2$, то в качестве x_{i+1} выбираем реализацию случайной величины, принимающей значения c_k с вероятностями q_k^1, $k = 1, \ldots, n$. Затем вычисляем $y_{i+1} = f(x_{i+1})$.

Если $q_1^1 > 1/2$, $b_1 \neq 0$, $d_1 \neq 1$, то вычисляем величины ξ_1, η_1, $q_1^2, \ldots, q_n^2$, $\bar{q}_1^2$. В качестве x_{i+1} выбираем реализацию случайной величины, принимающей значения c_j с

вероятностями q_j^2, $j=2, \ldots, n$, значение c_1 — с вероятностью $\bar{q}_1^2$, значения ξ_1, η_1 — каждое с вероятностью $(q_1^2-\bar{q}_1^2)/2$. Затем вычисляем $y_{i+1}=f(x_{i+1})$.

Если $q_1^1>1/2$ и, кроме того, $b_1=0$ или $d_1=1$, то вычисляем величины $\zeta_1, q_1^3, \ldots, q_n^3, \bar{q}_1^3$. В качестве x_{i+1} выбираем реализацию случайной величины, принимающей значения c_j с вероятностями q_j^3, $j=2, \ldots, n$, значение c_1 — с вероятностью $(q_1^3+\bar{q}_1^3)/2$, значение ζ_1 — с вероятностью $(q_1^3-\bar{q}_1^3)/2$. Затем вычисляем $y_{i+1}=f(x_{i+1})$.

Правило остановки может задаваться по-разному, например, в виде (4.20) гл. 1.

Обозначим построенный алгоритм через A.

Теорема 6.1 [250]. *Алгоритм A оптимален на один шаг на функциональном классе* (6.1) *среди стохастических последовательных алгоритмов поиска глобального максимума.* △

Доказательство теоремы дается леммами 6.1—6.8.

Отметим, что перед $(i+1)$-м шагом гарантированная в среднем априорная точность равна $Mv^1/2$ в случае $q_1^1\leqslant 1/2$; $Mv^2/2$ — в случае $q_1^1>1/2$, $b_1\neq 0$, $d_1\neq 1$; $Mv^3/2$ — в случае, когда $q>1/2$ и, кроме того, $b_1=0$ или $d_1=1$.

Результаты сравнения оптимального на один шаг стохастического алгоритма (с правилом остановки (4.20) гл. 1) и последовательно-оптимального по числу информационных вычислений детерминированного алгоритма (см. п. 3 § 4) на тестовых функциях вида $\sum_{k=1}^{9}\left(a_k\cos\frac{k\pi x}{100}+b_k\sin\frac{k\pi x}{100}\right)$, $x\in[0, 100]$, где a_k, b_k — независимые случайные величины с равномерным законом распределения, показывают некоторое преимущество стохастического алгоритма над детерминированным.

ГЛАВА 5

НЕКОТОРЫЕ СПЕЦИАЛЬНЫЕ КЛАССЫ ЭКСТРЕМАЛЬНЫХ ЗАДАЧ

В этой главе изучаются оптимальные методы решения нелинейных уравнений и систем уравнений, отыскания максимума функции минимума при наличии связей между переменными, а также оптимальные методы решения многокритериальных задач. Все эти задачи в основном вкладываются в рамки модели вычислений гл. 1 и сводятся к различным экстремальным задачам специального вида.

§ 1. Решение уравнений и систем уравнений

Методы решения нелинейных уравнений вида $f(x)=0$ и систем уравнений вида $f_i(x)=0$, $i=1, \ldots, n$, с достаточной полнотой изложены в учебной, справочной и монографической литературе [22, 27, 105, 116, 192, 196, 274, 276]. Во многих работах изучаются вопросы сложности, эффективности и оптимальности в различных классах алгоритмов, особенно в классе итерационных алгоритмов. Исследования сложности итерационных алгоритмов были инициированы монографией [274], хороший обзор с аннотированной библиографией имеется в [276]. Важнейшее значение в задачах решения уравнений имеют вопросы устойчивости решения в зависимости от исходных данных и вопросы регуляризации [79, 120, 178, 261, 268—271]. Среди работ, в которых рассматривается решение уравнений и систем уравнений и которые наиболее тесно связаны с общей моделью вычислений гл. 1, отметим [26, 42—45, 113, 146, 147, 158, 275, 276, 290, 323, 324, 340, 349, 355, 363, 372, 391, 429, 430, 436, 446].

1. Последовательно-оптимальный алгоритм поиска корня функции, принимающей на концах отрезка значения разных знаков. Поставим задачу отыскания какого-либо из корней функции f, удовлетворяющей на отрезке $K=[a, b]$ условию Липшица с известной вычислителю константой M и принимающей в точках a, b значения

разных знаков. Пусть для определенности

$$f \in F_1 \stackrel{\text{def}}{=} \{f \mid |f(u) - f(v)| \leqslant M|u - v|,\ u, v \in [a, b],\ f(a) \leqslant 0,\ f(b) \geqslant 0\}.$$

Очевидно, любая функция из класса F_1 имеет на сегменте $[a, b]$ хотя бы один корень. Будем предполагать, что информационными вычислениями являются вычисления значений функции f и что $B = K$, $S(f) = \{x_* \in K | f(x_*) = 0\}$ (тем самым мы несколько расширяем предположения § 1 гл. 1, допуская в качестве возможных значений $S(f)$ не только точки множества B, но и подмножества этого множества). Если считать, что $\gamma(S(f), \beta)$ есть расстояние от точки β, приближающей искомый корень, до множества $S(f)$, то стандартными рассуждениями (см., например, [276]) можно показать, что последовательно-оптимальным алгоритмом является *алгоритм бисекции* (алгоритм деления пополам на каждом шаге интервала локализации корня):

$$x_{i+1} = \frac{a_i + b_i}{2}, \quad i = 0, 1, \ldots, \tag{1.1}$$

где

$$a_0 = a, \quad b_0 = b,$$

$$a_{i+1} = \begin{cases} a_i, & y_{i+1} > 0, \\ x_{i+1} - y_{i+1}/M, & y_{i+1} < 0, \end{cases}$$

$$b_{i+1} = \begin{cases} x_{i+1} - y_{i+1}/M, & y_{i+1} > 0, \\ b_i, & y_{i+1} < 0; \end{cases}$$

если $y_{i+1} = 0$, то вычисления прекращаются — корень найден.

Мы рассмотрим далее иной критерий оценки эффективности, называемый иногда *критерием невязки*. Пусть F — класс функций, имеющих на K хотя бы один корень. Если $\beta \in K$ выбирается в качестве приближения к корню функции f, то *невязкой решения* назовем величину $|f(\beta)|$. Вычислитель стремится к минимизации невязки и, следовательно, задача решения уравнения сводится к экстремальной задаче специального вида — задаче минимизации функции с известным (нулевым) значением.

Эффективность вычислительных алгоритмов будем оценивать с помощью критерия

$$\gamma(S(f), \beta) = |f(\beta)|, \tag{1.2}$$

при этом

$$\varepsilon_N(z^N) = \inf_{\beta \in K} \max_{f \in F(z^N)} |f(\beta)|, \tag{1.3}$$

а в качестве итоговой операции используется функция $\widetilde{\beta}_*$, определенная формулами (4.3), (4.4) гл. 1. Таким образом, за корень функции f принимается точка $\widetilde{\beta}_*(z^N)$, реализующая нижнюю грань в (1.3). Заметим, что в рассматриваемой постановке введенная формулой (1.2) функция γ не является метрикой в B, поэтому $\widetilde{\beta}_*(z^N)$ нельзя трактовать как чебышевский центр множества неопределенности (ср. с п. 5 § 4 гл. 1). Несмотря на это, мы будем по-прежнему называть $\widetilde{\beta}_*$ центральной итоговой операцией.

Для произвольной функции $f \in F_1(z^i)$, $i \geqslant 1$, наряду с неравенством $\varphi_{1i}(x) \leqslant f(x) \leqslant \varphi_{2i}(x)$, $x \in [a, b]$, где

$$\varphi_{1i}(x) = \max_{j=1,\ldots,i} \{y_j - M|x - x_j|\},$$

$$\varphi_{2i}(x) = \min_{j=1,\ldots,i} \{y_j + M|x - x_j|\},$$

выполняется также неравенство $M(x-b) \leqslant f(x) \leqslant M(x-a)$, $x \in [a, b]$ (в противном случае оказались бы невыполненными условия $f(a) \leqslant 0$, $f(b) \geqslant 0$). Следовательно, $\varphi_{3i}(x) \leqslant f(x) \leqslant \varphi_{4i}(x)$, $x \in [a, b]$, где

$$\varphi_{3i}(x) = \max\{\varphi_{1i}(x), M(x-b)\},$$

$$\varphi_{4i}(x) = \min\{\varphi_{2i}(x), M(x-a)\}.$$

Заметим также, что для произвольной реализуемой ситуации z^i через любую точку $(x, y) \in \{(x, y) \mid a \leqslant x \leqslant b, \varphi_{3i}(x) \leqslant y \leqslant \varphi_{4i}(x)\}$ проходит график хотя бы одной функции из $F_1(z^i)$. Поэтому $\max\limits_{f \in F_1(z^N)} |f(\beta)| = \max\{|\varphi_{3N}(\beta)|, |\varphi_{4N}(\beta)|\}$, и выражение (1.3) принимает вид

$$\varepsilon_N(z^N) = \min_{\beta \in [a,b]} \max\{|\varphi_{3N}(\beta)|, |\varphi_{4N}(\beta)|\}. \tag{1.4}$$

Оказывается, что и в рассматриваемом случае (1.2), (1.3) алгоритм бисекции последовательно-оптимален по точности на классе F_1.

Теорема 1.1 [243]. *Алгоритм* (1.1) *последовательно-оптимален по точности на классе* F_1 *при произвольном заданном числе* N *информационных вычислений и*

гарантирует результат $M(b-a)\gamma_N$, *где* $\gamma_N = 1/(2^{N+2}-2)$. *Центральная итоговая операция* $\tilde{\beta}_*$ *определяется следующим условием: если* j*-е из чисел* $|y_1|, \ldots, |y_N|$, $M(b_N - a_N)/2$ *минимально среди этих чисел, то* $\tilde{\beta}_*(z^N)$ — *это* j*-я точка из множества* $\{x_1, \ldots, x_N, (a_N + b_N)/2\}$.

Доказательство. Покажем, что алгоритм (1.1) оптимален по точности в множестве всех последовательных алгоритмов $\tilde{X}^N$, а наилучший гарантированный результат равен $M(b-a)\gamma_N$.

Убедимся сначала, что алгоритм (1.1) гарантирует результат не хуже $M(b-a)\gamma_N$. Пусть $f \in F_1$, точки $x_1, \ldots, x_N$ получены способом (1.1), $y_j = f(x_j)$, $j = 1, \ldots, N$.

Если $|y_j| \leqslant M(b-a)\gamma_N$ хотя бы для одного $j \in \{1, \ldots, N\}$, то $|\varphi_{3N}(x_j)| = |\varphi_{4N}(x_j)| = |y_j| \leqslant M(b-a)\gamma_N$, откуда в силу (1.4) $\varepsilon_N(z^N) \leqslant M(b-a)\gamma_N$.

Пусть теперь

$$|y_j| > M(b-a)\gamma_N, \quad j = 1, \ldots, N. \tag{1.5}$$

Покажем, что в этом случае

$$b_j - a_j < (b-a)\left(\frac{1}{2^j} - \frac{2^j - 1}{2^{j-1}}\gamma_N\right), \quad j = 1, \ldots, N. \tag{1.6}$$

Действительно, из (1.1) и (1.5) следует, что

$$b_1 - a_1 = \frac{b-a}{2} - \frac{|y_1|}{M} < \frac{b-a}{2} - (b-a)\gamma_N = \\ = (b-a)\left(\frac{1}{2} - \gamma_N\right).$$

Предположим, что (1.6) выполняется для $j = i-1$, $i \leqslant N$. Тогда

$$b_i - a_i = \frac{b_{i-1} - a_{i-1}}{2} - \frac{|y_i|}{M} < \frac{b-a}{2}\left(\frac{1}{2^{i-1}} - \frac{2^{i-1}-1}{2^{i-2}}\gamma_N\right) - \\ - (b-a)\gamma_N = (b-a)\left(\frac{1}{2^i} - \frac{2^i - 1}{2^{i-1}}\gamma_N\right),$$

и неравенство (1.6) доказано, таким образом, по индукции. Из него, в частности, вытекает, что

$$M\frac{b_N - a_N}{2} < M(b-a)\left(\frac{1}{2^{N+1}} - \frac{2^N - 1}{2^N}\gamma_N\right) = M(b-a)\gamma_N.$$

Отсюда, учитывая (1.4), получаем (см. рис. 16)

$$\varepsilon_N(z^N) \leqslant \max\left\{\left|\varphi_{3N}\left(\frac{a_N+b_N}{2}\right)\right|, \left|\varphi_{4N}\left(\frac{a_N+b_N}{2}\right)\right|\right\} = \\ = M\frac{b_N-a_N}{2} < M(b-a)\gamma_N.$$

Итак, мы убедились в том, что алгоритм (1.1) гарантирует результат не хуже $M(b-a)\gamma_N$. Кроме того, нетрудно усмотреть (рис. 16), что минимум в (1.4) достигается

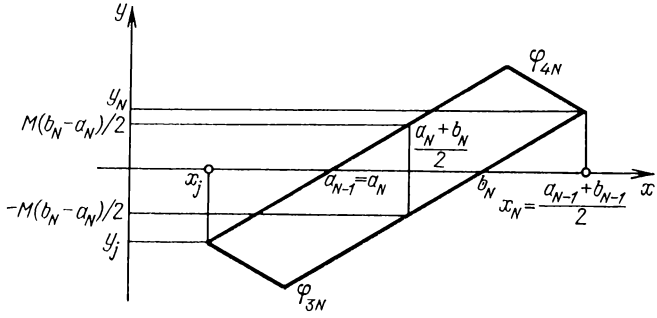

Рис. 16. К вопросу об определении величины $\varepsilon_N(z^N)$ для класса F_1

в одной из точек $x_1, \ldots, x_N$, $(a_N+b_N)/2$ и равен

$$\varepsilon_N(z^N) = \min\left\{|y_1|, \ldots, |y_N|, M\frac{b_N-a_N}{2}\right\}. \qquad (1.7)$$

За приближение $\tilde{\beta}_*(z^N)$ к корню, следовательно, надо принять j-ю из точек $x_1, \ldots, x_N$, $(a_N+b_N)/2$, если j-е из чисел $|y_1|, \ldots, |y_N|$, $M(b_N-a_N)/2$ минимально среди этих чисел.

Убедимся теперь в том, что гарантировать результат лучше $M(b-a)\gamma_N$ невозможно, т. е. для произвольного последовательного алгоритма $\tilde{x}^N \in \tilde{X}^N$ существует такая функция $f \in F_1$, что $\varepsilon_N(x^N, y^N) \geqslant M(b-a)\gamma_N$, где вектор x^N получен по формулам (3.2) гл. 1 при применении алгоритма $\tilde{x}^N$ к функции f, а $y_j = f(x_j)$, $j = 1, \ldots, N$. Для этого достаточно рассмотреть функцию f, для которой $|y_j| = M(b-a)\gamma_N$, $j = 1, \ldots, N$, а знаки поочередно вычисляемых чисел $y_1, \ldots, y_N$ подчиняются следующей закономерности. Если знаки значений функции f в ближайшей к x_j слева и в ближайшей к x_j справа среди

точек a, x_1, ..., x_{j-1}, b совпадают, то и знак y_j такой же. Если же эти знаки разные, т. е. x_j лежит в сегменте локализации корня (этот сегмент обозначим $[a_{j-1}, b_{j-1}]$), то знак y_j таков, что длина следующего сегмента $[a_j, b_j]$ локализации корня определяется условием $b_j - a_j = \max\{b_{j-1} - x_j, x_j - a_{j-1}\} - (b-a)\gamma_N$. Легко видеть, что такая функция $f \in F_1$ существует и для нее $\varepsilon_N(x^N, y^N) = M(b-a)\gamma_N$. Тем самым оптимальность алгоритма (1.1) в множестве всех последовательных алгоритмов $\tilde{X}^N$ доказана. Докажем, что этот алгоритм последовательно-оптимален по точности.

Пусть значения функции вычислены в точках x_1, ..., x_i, $i < N$, и равны соответственно y_1, ..., y_i; $[a_i, b_i]$ — сегмент локализации корня, и осталось вычислить $N-i$ значений функции f. Тогда, используя уже доказанное, получаем, что наилучший гарантированный результат в рассматриваемой ситуации z^i равен

$$\min\{|y_1|, \ldots, |y_i|, M(b_i - a_i)\gamma_{N-i}\}. \tag{1.8}$$

Ясно, что результат (1.8) можно гарантировать с помощью алгоритма (1.1). Следовательно, алгоритм (1.1) последовательно-оптимален. △

Замечание 1. Если $\min\{|y_1|, \ldots, |y_i|\} \leqslant M(b_i - a_i) \times \gamma_{N-i}$, то гарантировать улучшение результата оставшимися $N-i$ вычислениями нельзя. Поэтому выбор точки x_{i+1} произволен. Этот произвол можно использовать для улучшения наилучшего гарантированного результата, выбирая точку x_{i+1} случайно, т. е. применяя стохастические алгоритмы. Например, если $x_{i+1} \in [a_i, b_i]$ — случайная величина с равномерным законом распределения, то гарантированный в среднем результат лучше результата $\min\{|y_1|, \ldots, |y_i|\}$, который можно гарантировать детерминированными алгоритмами.

Замечание 2. Выбор точки очередного вычисления алгоритмом (1.1) не зависит от заданного числа вычислений N. Следовательно, алгоритм (1.1) последовательно-оптимален по числу информационных вычислений при любой заданной точности ε. При этом вычисления следует прекратить после N шагов, если

$$\varepsilon_N(z^N) = \min\{|y_1|, \ldots, |y_N|, M(b_N - a_N)/2\} \leqslant \varepsilon. \ \triangle$$

2. Оптимальные по точности на функциональном классе F_2 алгоритмы. Обозначим через F_2 класс функций, удовлетворяющих на сегменте $[a, b]$ условию Липшица с

известной вычислителю константой M и имеющих на $[a, b]$ хотя бы один корень. Нетрудно видеть, что $F_2 \supset F_1$.

Построим оптимальные по точности алгоритмы, предполагая, что критерием оценки эффективности служит критерий невязки (1.3).

Пусть произведено N вычислений, $y_j = f(x_j)$, $j = 1, \ldots, N$. Могут представиться следующие три случая:

А) знаки не всех чисел $y_1, \ldots, y_N$ одинаковы;

Б) знаки всех чисел $y_1, \ldots, y_N$ одинаковы, множество

$$[a, b] \setminus \bigcup_{j=1}^{N} (x_j - |y_j|/M,\ x_j + |y_j|/M)$$

состоит не менее чем из двух непересекающихся сегментов (только в точках этого множества функции $f \in F_2(z^N)$ могут принимать нулевые значения; оно непусто, так как любая функция $f \in F_2$ имеет хотя бы один корень);

В) знаки всех чисел $y_1, \ldots, y_N$ одинаковы,

$$[a, b] \setminus \bigcup_{j=1}^{N} (x_j - |y_j|/M,\ x_j + |y_j|/M) = [a_N, b_N].$$

В случаях А) и Б) через любую точку

$$(x,\ y) \in \{(x,\ y) \mid a \leqslant x \leqslant b,\ \varphi_{1N}(x) \leqslant y \leqslant \varphi_{2N}(x)\}$$

проходит график хотя бы одной функции $f \in F_2(z^N)$, поэтому

$$\varepsilon_N(z^N) = \min_{\beta \in [a,b]} \max\{|\varphi_{1N}(\beta)|, |\varphi_{2N}(\beta)|\}, \tag{1.9}$$

при этом в случае Б)

$$\varepsilon_N(z^N) = \min_{j=1,\ldots,N} |y_j|. \tag{1.10}$$

В случае В) любая функция $f \in F_2(z^N)$ имеет на $[a_N, b_N]$ корень и потому

$$\min\{-M(x - a_N),\ M(x - b_N)\} \leqslant f(x) \leqslant$$
$$\leqslant \max\{M(x - a_N),\ -M(x - b_N)\}.$$

Объединяя это неравенство с неравенством $\varphi_{1N}(x) \leqslant f(x) \leqslant \varphi_{2N}(x)$, получим, что $\varphi_{5N}(x) \leqslant f(x) \leqslant \varphi_{6N}(x)$, где

$$\varphi_{5N}(x) = \max\{\varphi_{1N}(x),\ \min[-M(x - a_N),\ M(x - b_N)]\},$$
$$\varphi_{6N}(x) = \min\{\varphi_{2N}(x),\ \max[M(x - a_N),\ -M(x - b_N)]\}.$$

Поэтому в случае В) (см. рис. 17)

$$\varepsilon_N(z^N) = \min_{\beta\in B} \max\{|\varphi_{5N}(\beta)|,\ |\varphi_{6N}(\beta)|\} =$$

$$= \min\left\{|y_1|, \ldots, |y_N|,\ M\frac{b_N - a_N}{2}\right\}. \quad (1.11)$$

Решение вопроса о центральной итоговой операции $\tilde{\beta}_*$ не представляет труда. После окончания N информационных вычислений за приближение $\tilde{\beta}_*(z^N)$ к корню функции $f \in F_2(z^N)$ в случаях А) и В) следует принять

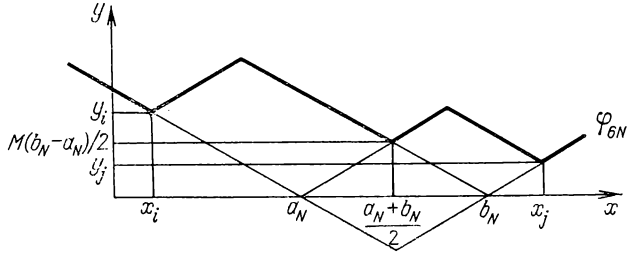

Рис. 17. К вопросу об определении величины $\varepsilon_N(z^N)$ для класса F_2 в случае В)

j-ю из точек $x_1, \ldots, x_N$, $(a_N + b_N)/2$, если j-е из чисел $|y_1|, \ldots, |y_N|$, $(b_N - a_N)/2$ минимально среди этих чисел, а в случае Б) — j-ю из точек $x_1, \ldots, x_N$, если j-е из чисел $|y_1|, \ldots, |y_N|$ минимально среди них. При этом под $[a_N, b_N]$ в случае А) мы понимаем сегмент локализации корня наименьшей длины (если таких сегментов несколько). Если этим условием $[a_N, b_N]$ не определяется однозначно, то для определенности будем считать, что $[a_N, b_N]$ — самый левый из возможных сегментов.

Будем считать, что в качестве итоговой операции применяется центральная итоговая операция $\tilde{\beta}_*$. Имеет место следующее утверждение.

Теорема 1.2 [243]. *Пассивные алгоритмы*

$$x^N(i) = (x_1^0, \ldots, x_{i-1}^0, x_{i+1}^0, \ldots, x_{N+1}^0), \quad i = 1, \ldots, N+1,$$

где

$$x_j^0 = a + \frac{2j-1}{2(N+1)}(b-a), \quad j = 1, \ldots, N+1,$$

оптимальны по точности на классе функций F_2 в множе-

стве $\tilde{X}^N$ всех последовательных алгоритмов и гарантируют результат $v \overset{\text{def}}{=} M(b-a)/[2(N+1)]$.

Доказательство. Покажем, что алгоритм $x^N(i)$ гарантирует результат v, $i = 1, \ldots, N+1$. Пусть $y_j^0 = |f(x_j^0)| > v$, $j = 1, \ldots, i-1, i+1, \ldots, N+1$, $f \in F_2$. Тогда имеет место либо случай А), причем $\operatorname{sign} y_{i-1}^0 \times \operatorname{sign} y_{i+1}^0 = -1$, либо случай В). В обоих случаях $[a_N, b_N] = [x_{i-1}^0 + |y_{i-1}^0|/M,\ x_{i+1}^0 - |y_{i+1}^0|/M]$, и согласно (1.7), (1.11)

$$\varepsilon_N(z^N) \leqslant M\frac{b_N - a_N}{2} \leqslant \frac{M}{2}\left[\frac{2(b-a)}{N+1} - \frac{2v}{M}\right] = v,$$

где $z^N = (x^N(i), y_1^0, \ldots, y_{i-1}^0, y_{i+1}^0, \ldots, y_{N+1}^0)$.

Для завершения доказательства теоремы достаточно убедиться, что никакой алгоритм $\tilde{x}^N \in \tilde{X}^N$ не может гарантировать результат лучше v. Покажем, что для любого алгоритма $\tilde{x}^N \in \tilde{X}^N$ существует такая функция $f \in F_2$, что $\varepsilon_N(x^N, y^N) = v$, где вектор x^N получен по формулам (3.2) гл. 1 при применении алгоритма $\tilde{x}^N$ к функции f, а $y_j = f(x_j)$, $j = 1, \ldots, N$.

Удостоверимся сначала, что для произвольного вектора $(x_1, \ldots, x_N)$ такого, что $a \leqslant x_1 \leqslant \ldots \leqslant x_N \leqslant b$, либо выполняются хотя бы два из $N+1$ неравенств

$$x_1 - a \geqslant \frac{b-a}{2(N+1)},\quad x_{j+1} - x_j \geqslant \frac{b-a}{N+1},\quad j = 1, \ldots, N-1,$$
$$b - x_N \geqslant \frac{b-a}{2(N+1)}, \tag{1.12}$$

либо справедливо хотя бы одно из трех неравенств

$$x_1 - a > \frac{3(b-a)}{2(N+1)}, \tag{1.13}$$

$$x_{k+1} - x_k > \frac{2(b-a)}{N+1},\quad k \in \{1, \ldots, N-1\}, \tag{1.14}$$

$$b - x_N > \frac{3(b-a)}{2(N+1)}. \tag{1.15}$$

Действительно, предположим, что выполняется не более чем одно из неравенств (1.12). Это может быть в трех случаях:

$$x_{j+1} - x_j < \frac{b-a}{N+1},\ j = 1, \ldots, N-1,\ b - x_N < \frac{b-a}{2(N+1)}; \tag{1.16}$$

$$x_1 - a < \frac{b-a}{2(N+1)}, \quad x_{j+1} - x_j < \frac{b-a}{N+1},$$
$$j = 1, \ldots, k-1, k+1, \ldots, N-1,$$
$$k \in \{1, \ldots, N-1\}, \quad b - x_N < \frac{b-a}{2(N+1)}; \tag{1.17}$$
$$x_1 - a < \frac{b-a}{2(N+1)}, \quad x_{j+1} - x_j < \frac{b-a}{N+1}, \quad j = 1, \ldots, N. \tag{1.18}$$

В случае (1.16), сложив все N неравенств, получим

$$(b-a) - (x_1 - a) < \frac{(2N-1)(b-a)}{2(N+1)},$$

откуда следует (1.13). Аналогично, из (1.17) следует (1.14), а из (1.18)—(1.15).

Рассмотрим теперь вектор $x^N = (x_1, \ldots, x_N)$, который получается по формулам (3.2) гл. 1 для алгоритма $\tilde{x}^N$ при $y_1 = \ldots = y_N = v$. Нетрудно видеть, что $y_1 = \ldots = y_j = v$ образуют реализуемую ситуацию при любых $x_1, \ldots, x_j$, поэтому значения $\tilde{x}_{j+1}(z^j)$ определены, $j = 2, \ldots, N-1$. Без ограничения общности можно считать, что $a \leqslant x_1 \leqslant \ldots \leqslant x_N \leqslant b$. Положим $f(x) = \max\limits_{j=1,\ldots,n} \{v - M|x - x_j|\}$. В силу леммы 2.7 гл. 1, функция f удовлетворяет условию Липшица с константой M. Она имеет корень на тех из $N+1$ промежутков $[0, x_1)$, (x_1, x_2), $\ldots$, (x_{N-1}, x_N), $(x_N, b]$, для которых соответствующие неравенства в (1.12) справедливы. Хотя бы один такой промежуток, вследствие ранее доказанного, имеется. Таким образом, $f \in F_2$. Нетрудно видеть, что $y_j = f(x_j) = v$, $j = 1, \ldots, N$, и, стало быть, при применении алгоритма $\tilde{x}^N$ к функции f получается определенный выше вектор x^N.

Если выполняются хотя бы два из неравенств (1.12), то имеет место случай Б) и, в силу (1.10), $\varepsilon_N(x^N, y^N) = v$. Если выполняется не более чем одно из неравенств (1.12), то имеет место случай В). При этом, если выполняется (1.13), то $b_N - a_N = x_1 - a - v/M > (b-a)/(N+1)$; если выполняется (1.14), то $b_N - a_N = x_{k+1} - x_k - 2v/M > (b-a)/(N+1)$; если, наконец, выполняется (1.15), то $b_N - a_N = b - x_N - v/M > (b-a)/(N+1)$. Таким образом, и в случае В) (см. (1.11)) $\varepsilon_N(x^N, y^N) = \min\{v, M(b_N - a_N)/2\} = v$. Итак, в обоих возможных случаях $\varepsilon_N(x^N, y^N) = v$. $\triangle$

Отметим, что N-шаговый оптимальный алгоритм состоит, согласно доказанной теореме, в вычислении значе-

ний функции в N центрах оптимального покрытия сегмента $[a, b]$ равными сегментами, число которых равно $N+1$.

3. Последовательно-оптимальный по точности на функциональном классе F_2 алгоритм. Опишем вкратце построение последовательно-оптимального по точности алгоритма, предполагая по-прежнему, что критерием оценки эффективности служит критерий невязки (1.2) (обоснования, аналогичные проведенным при доказательстве теоремы 1.2, опускаем).

Точкой первого информационного вычисления для последовательно-оптимального алгоритма может служить любая из точек, возможных для оптимальных алгоритмов, т. е.

$$x_1 \in \left\{a + \frac{2j-1}{2(N+1)}(b-a) \mid j = 1, \ldots, N+1\right\}.$$

Пусть произведено i информационных вычислений, $i \geqslant 1$, $\operatorname{sign} y_1 = \ldots = \operatorname{sign} y_i$. Положим $m_i = \min\{|y_1|, \ldots, |y_i|\}$. Обозначим через r_{iN} радиус оптимального покрытия $N-i+1$ равными сегментами множества

$$[a, b] \setminus \bigcup_{j=1}^{i} (x_j - |y_j|/M,\ x_j + |y_j|/M). \tag{1.19}$$

Наилучший гарантированный в ситуации z^i результат равен $\min\{Mr_{iN}, m_i\}$. Если $Mr_{iN} < m_i$, то x_{i+1} — любой из центров оптимального покрытия множества (1.19). Если же $Mr_{iN} \geqslant m_i$, то гарантировать улучшение результата оставшимися $N-i$ вычислениями нельзя. Поэтому выбор точки x_{i+1} произволен.

Если же допустимо применение стохастических алгоритмов, то так же, как и для класса F_1 в случае $\min\{|y_1|, \ldots, |y_i|\} \leqslant M(b_i - a_i)\gamma_{N-i}$ (ср. с замечанием 1 к теореме 1.1), наилучший гарантированный детерминированными алгоритмами результат можно улучшить в среднем, выбирая точку x_{i+1} случайно.

Алгоритм отыскания числа r_{iN} описан в [242, 243], он аналогичен алгоритму отыскания R_{iN} (см. п. 1 § 4, гл. 4).

Предположим теперь, что произведено i информационных вычислений, причем $\operatorname{sign} y_1 = \ldots = \operatorname{sign} y_{i-1} \neq \operatorname{sign} y_i$. Пусть для определенности $y_1, \ldots, y_{i-1} > 0$, $y_i < 0$. Пусть точка x_{i_1} — ближайшая из точек $x_1, \ldots, x_{i-1}$ к точке x_i слева, x_{i_2} — справа (одна из точек x_{i_1}, x_{i_2} может не существовать, так как $x_1, \ldots, x_{i-1}$ могут лежать

по одну сторону от x_i). За $[a_i, b_i]$ принимается тот из сегментов $[x_{i_1} + y_{i_1}/M, x_i + y_i/M]$, $[x_i - y_i/M, x_{i_2} - y_{i_2}/M]$, длина которого меньше. Точка x_{i+1} и последующие выбираются в соответствии с алгоритмом (1.1).

4. Решение системы уравнений. Рассмотрим систему уравнений

$$f_i(x) = 0, \quad i = 1, \ldots, m, \quad x \in K \subset \mathbf{R}^n, \tag{1.20}$$

предполагая, что $f_i \in F_\rho$, $i = 1, \ldots, m$ (где ρ — некоторая квазиметрика), и что хотя бы одно решение системы (1.20) существует. Задача отыскания какого-либо решения системы (1.20) эквивалентна задаче минимизации функции невязки $\max\limits_{i=1,\ldots,m} |f_i(x)|$, $x \in K$, или, что то же самое, задаче максимизации функции $f(x) = -\max\limits_{i=1,\ldots,m} |f_i(x)|$ на множестве K. В силу леммы 2.7 гл. 1 функция f принадлежит классу F_ρ, поэтому задачу можно приближенно решить с помощью алгоритмов, разработанных в § 3 гл. 4. При этом следует, однако, учесть, что максимальное значение функции f известно и равно нулю. Это обстоятельство позволяет несколько модифицировать алгоритмы из § 3 гл. 4, что приводит к упрощению алгоритмов, повышая в то же время их эффективность.

Пусть фиксирована точность ε, которую следует обеспечить при решении задачи максимизации функции f, т. е. необходимо указать точку x_ε такую, что $f(x_\varepsilon) \geqslant -\varepsilon$. Ясно, что это можно сделать при помощи модификации алгоритма, описанного в п. 3 § 3 гл. 4. Модификация состоит в замене во всех построениях множеств $CK_i(\bar{y}_i + \varepsilon)$ множествами $CK_i(-\varepsilon)$.

Алгоритм из п. 3 § 3 гл. 4 и предложенную его модификацию можно применять и в случае, когда система (1.20) не предполагается совместной. При этом, если после i шагов условие остановки $CK_i(\bar{y}_i + \varepsilon) = \varnothing$ для немодифицированного алгоритма оказывается выполненным и в то же время $\bar{y}_i < -\varepsilon$, или же $CK_i(-\varepsilon) = \varnothing$, $\bar{y}_i < -\varepsilon$ для модифицированного алгоритма, то это означает, что система (1.20) несовместна.

При $n > 2$ даже в сравнительно простом случае (4.13) гл. 4 непосредственная реализация описанных алгоритмов по причинам, упомянутым в п. 6 § 4 гл. 4, невозможна. В этом случае решение задачи можно получить с помощью алгоритма (4.14), (4.15) гл. 4. Модификация этого алгоритма по тому же, что и выше, способу позво-

ляет упростить его и повысить эффективность. Выбор точки $(i+1)$-го информационного вычисления модифицированный алгоритм производит по правилу (ср. с (4.15) гл. 4)

$$x_{i+1} \in \{u_j \mid 1 \leqslant j \leqslant N,\ \varphi_{2i}(u_j) \geqslant -\varepsilon\}. \tag{1.21}$$

Данный алгоритм целесообразно дополнить блоком локального уточнения решения. В [46] блок локального уточнения состоит в том, что решение, полученное с помощью алгоритма (1.21), выбирается в качестве начальной точки для одного из методов ньютоновского типа, предложенного в [327, 328].

§ 2. Отыскание максимума функции минимума при связанных переменных

Применение принципа наилучшего гарантированного результата в исследовании операций и теории игр приводит к необходимости решения разнообразных минимаксных и максиминных задач [62, 63]. Простейшей из них является задача нахождения максимина с распадающимися переменными [90, 286]

$$\max_{x \in X} \min_{v \in V} Q(x, v). \tag{2.1}$$

В этом параграфе мы рассмотрим более общую, чем (2.1), максиминную задачу со связанными переменными:

$$f(x) \stackrel{\text{def}}{=} \min_{v \in B(x)} Q(x, v) \to \sup_{x \in K}, \tag{2.2}$$

где

$$B(x) = \{v \in V \mid \varphi_j(x, v) \geqslant 0, \quad j = 1, \ldots, m\}, \quad x \in X,$$

и $K = \{x \in X \mid B(x) \neq \varnothing\}$, V — метрическое пространство. Задачи типа (2.2) распространены в теории иерархических систем и игр с обменом информацией [63].

1. Об условиях принадлежности функции минимума к функциональному классу, заданному квазиметрикой. Легко построить примеры (см., например, [286]), когда функция минимума вида (2.2) разрывна при гладких Q, $\varphi_1, \ldots, \varphi_m$. В таких условиях максимизация этой функции затруднена. Наша цель — найти условия на Q, $\varphi_1, \ldots, \varphi_m$, гарантирующие наличие у функции f свойств, позволяющих строить алгоритмы отыскания ее глобального мак-

симума. Так, если

$$|f(x_1)-f(x_2)| \leqslant \rho(x_1, x_2), \quad x_1, x_2 \in K, \tag{2.3}$$

и ρ обладает свойствами квазиметрики, то задачу (2.2) можно решать с помощью алгоритмов, разработанных в гл. 4. В связи с этим нас будут интересовать условия, при которых (2.3) выполнено, а функция ρ в (2.3) обладает свойствами квазиметрики.

Начнем с установления некоторых важных свойств функции минимума f, определенной формулой (2.2).

Предположим, что выполнены следующие условия:

$$|Q(x_1, v)-Q(x_2, v)| \leqslant \alpha(x_1, x_2), \quad x_1, x_2 \in X, \quad v \in V, \tag{2.4}$$

где α — квазиметрика;

$$|\varphi_j(x_1, v)-\varphi_j(x_2, v)| \leqslant \beta(x_1, x_2), \quad x_1, x_2 \in X, \quad v \in V, \quad j=1, \ldots, m, \tag{2.5}$$

где β — квазиметрика;

$$|Q(x, v_1)-Q(x, v_2)| \leqslant \gamma(r(v_1, v_2)), \; x \in X, \; v_1, v_2 \in V, \tag{2.6}$$

где функция γ определена и не убывает на $[0, \infty)$, $\gamma(0)=0$, r — расстояние в V;

$$\min_{j=1,\ldots,m} \varphi_j(x, v) \leqslant -\omega(r(v, B(x))) \text{ при } v \notin B(x), \tag{2.7}$$

где функция ω определена, непрерывна и возрастает на $[0, \infty)$, $\omega(0)=0$, $\lim\limits_{r\to\infty} \omega(r)=\infty$, $r(v, A)=\inf\limits_{\bar{v}\in A} r(v, \bar{v})$ — расстояние от точки v до множества A.

Теорема 2.1 [257]. *Если выполнены условия* (2.4)—(2.7), *то функция* f *вида* (2.2) *удовлетворяет соотношению* (2.3) *с*

$$\rho(x_1, x_2)=\alpha(x_1, x_2)+\gamma[\omega^{-1}(\beta(x_1, x_2))], \tag{2.8}$$

где ω^{-1} — *функция, обратная* ω.

Доказательство. В силу (2.5) и (2.7) при произвольных $x_1, x_2 \in X$, $v \notin B(x_2)$ справедливы неравенства

$$\min_{1\leqslant j\leqslant m} \varphi_j(x_1, v) \leqslant \min_{1\leqslant j\leqslant m} \varphi_j(x_2, v)+\beta(x_1, x_2) \leqslant$$
$$\leqslant -\omega(r(v, B(x_2)))+\beta(x_1, x_2).$$

Если $r(v, B(x_2))>\omega^{-1}(\beta(x_1, x_2))$, то $\min\limits_{1\leqslant j\leqslant m} \varphi_j(x_1, v)<0$, и, следовательно, $v \notin B(x_1)$.

Таким образом, множество $B(x_1)$ лежит в $\omega^{-1}(\beta(x_1, x_2))$ — окрестности множества $B(x_2)$, т. е. для любой точки $v_1 \in B(x_1)$ найдется такая точка $v_2 \in B(x_2)$, что $r(v_1, v_2) \leqslant \omega^{-1}(\beta(x_1, x_2))$. Поэтому

$$Q(x_2, v_2) - Q(x_1, v_1) = \\ = [Q(x_2, v_2) - Q(x_1, v_2)] + [Q(x_1, v_2) - Q(x_1, v_1)] \leqslant \\ \leqslant \alpha(x_1, x_2) + \gamma[\omega^{-1}(\beta(x_1, x_2))] = \rho(x_1, x_2),$$

откуда следует, что $f(x_2) \leqslant Q(x_1, v_1) + \rho(x_1, x_2)$ при произвольном $v_1 \in B(x_1)$, и потому $f(x_2) \leqslant f(x_1) + \rho(x_1, x_2)$. Аналогично устанавливается неравенство $f(x_1) \leqslant f(x_2) + \rho(x_1, x_2)$, что завершает доказательство. △

Предположим дополнительно, что V — линейное нормированное пространство с нормой $\|\cdot\|$,

$$r(v_1, v_2) = \|v_1 - v_2\|, \quad v_1, v_2 \in V, \tag{2.9}$$

функция γ обладает свойством

$$\gamma(r_1 + r_2) \leqslant \gamma(r_1) + \gamma(r_2), \quad r_1, r_2 \geqslant 0, \tag{2.10}$$

а функция ω непрерывна и обладает свойством

$$\omega(r_1 + r_2) \geqslant \omega(r_1) + \omega(r_2), \quad r_1, r_2 \geqslant 0. \tag{2.11}$$

Заметим, что выполнение свойства (2.11) для непрерывной возрастающей на $[0, \infty)$ функции ω влечет за собой выполнение свойства

$$\omega^{-1}(\omega_1 + \omega_2) \leqslant \omega^{-1}(\omega_1) + \omega^{-1}(\omega_2), \quad \omega_1, \omega_2 \geqslant 0. \tag{2.12}$$

Теорема 2.2 [257]. *Пусть выполнены условия* (2.4)—(2.7), (2.9)—(2.11). *Тогда функция* ρ, *заданная формулой* (2.8), *удовлетворяет свойствам квазиметрики и оценка* (2.3) *для функций вида* (2.2) *точна.*

Доказательство. Выполнение свойств квазиметрики для ρ непосредственно вытекает из сделанных предположений и неравенства (2.12), являющегося следствием этих предположений.

Убедимся теперь, что оценка (2.3) точна. Для произвольных $a \in K$, $v_0 \in V$ положим

$$Q(x, v) = -\alpha(a, x) - \gamma(r(v, v_0)), \tag{2.13}$$

$$\varphi_j(x, v) = \beta(a, x) - \omega(r(v, v_0)), \quad j = 1, \ldots, m, \tag{2.14}$$

где α, β, γ, ω удовлетворяют всем наложенным в теореме условиям. Тогда, как нетрудно проверить, (2.4)—(2.6) выполнены. Проверим выполнение (2.7). Пусть $v \notin$

$\not\in B(x)=\{v|r(v, v_0)\leqslant \omega^{-1}(\beta(a, x))\}$. Тогда (см. рис. 18)

$$r(v, v_0)=r(v, B(x))+\omega^{-1}(\beta(a, x)),$$

откуда, используя (2.11), получаем

$$\omega(r(v, v_0))=\omega[r(v, B(x))+\omega^{-1}(\beta(a, x))]\geqslant$$
$$\geqslant \omega(r(v, B(x)))+\beta(a, x),$$

а это и есть неравенство (2.7).

Вычислим теперь в случае (2.13), (2.14) функцию вида (2.2):

$$\min_{v\in B(x)} Q(x, v)=-\alpha(a, x)-\max_{v\in\{v|r(v,v_0)\leqslant\omega^{-1}(\beta(a,x))\}}\gamma(r(v, v_0))=$$
$$=-\alpha(a, x)-\gamma[\omega^{-1}(\beta(a, x))]=-\rho(a, x). \quad (2.15)$$

Положив в формулах (2.13), (2.14) $a=x_1$, для $f(x)=\min\limits_{v\in B(x)} Q(x, v)$ в силу (2.15) имеем

$$|f(x_1)-f(x_2)|=\rho(x_1, x_2). \ \triangle$$

2. Оптимальные алгоритмы максимизации функции минимума. Сформулируем теперь основной результат об оптимальных методах решения задачи (2.2). Будем предполагать, что условия (2.4)—(2.7), (2.9)—(2.11) выполнены. Через $\bar{F}$ обозначим подкласс функционального класса F_ρ, содержащий все функции вида (2.2). Через X^N и $\tilde{X}^N$ обозначаем, как обычно, соответственно множества пассивных и всех последовательных алгоритмов с N информационными вычислениями.

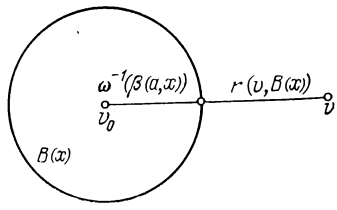

Рис. 18. К доказательству теоремы 2.2

Теорема 2.3 [257]. *Оптимальный по точности в X^N на классе F_ρ алгоритм отыскания глобального максимума оптимален по точности в X^N и в $\tilde{X}^N$ на подклассе $\bar{F}$; наилучшие гарантированные результаты на F_ρ и на $\bar{F}$ совпадают.*

Доказательство. При доказательстве теоремы 2.2 было установлено, что все функции

$$h_a(x)=-\rho(a, x), \quad a\in K, \quad (2.16)$$

принадлежат классу $\bar{F}$. Теперь оптимальность в X^N на

классе $\bar{F}$ алгоритма $x_0^N \in X^N$, оптимального на F_ρ, а также совпадение наилучших гарантированных результатов в X^N на F_ρ и на $\bar{F}$ следуют из замечания 1 к теореме 2.1 гл. 4.

Далее, нетрудно видеть, что класс $\bar{F}$ содержит все константы и удовлетворяет требованиям (5.14), (5.15) гл. 1. Выполнение, например, (5.15) гл. 1 следует из того, что при любом $x \in K$

$$f(x) = \min_{v \in B(x)} Q(x, v) \Rightarrow \max\{f(x), 0\} = \min_{v \in B(x)} \max\{Q(x, v), 0\}.$$

Поэтому, в силу теоремы 5.3 гл. 1, алгоритм x_0^N оптимален по точности в $\bar{X}^N$ на $\bar{F}$. $\triangle$

Полученные результаты в принципе решают задачу об оптимальном поиске максимума функций f вида (2.2). Соответствующие алгоритмы построены в гл. 4. Однако вопросы построения оптимальных покрытий множества K, возникающие при реализации оптимальных алгоритмов, так же как и вопросы оптимального вычисления значений $f(x)$, остаются открытыми. Впрочем, в некоторых случаях, например, в случае $B(x) = \{v \in V \mid \psi_j^-(x) \leqslant v \leqslant \psi_j^+(x),\ j = 1, \ldots, l\}$ задачу удается решить до конца. Пример решения конкретной задачи такого рода приведен в [257].

3. Об условиях теорем 2.1—2.3. Сделаем несколько замечаний по поводу условий теорем 2.1—2.3 и укажем на некоторые практически важные частные случаи этих условий.

Условиям (2.4), (2.5) удовлетворяют произвольные ограниченные функции, условию (2.6) при выполнении (2.9), (2.10) удовлетворяют, например, равномерно непрерывные функции с модулем непрерывности γ. Единственным в известной мере ограничительным условием является (2.7). Оно выполняется, например, с $\omega(r) = Rr$, $R > 0$ (условие регулярности) в следующих случаях [286]:

а) функции $\varphi_j(x, v)$ непрерывны на произведении компактов $X \times \bar{V}$, где $\bar{V} \supset V$, причем V — конечное множество, $j = 1, \ldots, m$;

б) функции $\varphi_j(x, v)$ линейны по v на многогранном множестве $V \subset \mathbf{R}^n$, т. е. $\varphi_j(x, v) = \langle a_j, v \rangle + b_j(x)$, $j = 1, \ldots, m$;

в) функции $\varphi_j(x, v)$ вогнуты по v на выпуклом компакте $V \subset \mathbf{R}^n$, непрерывны по (x, v) и удовлетворяют

условию Слейтера, т. е. для любого $x \in X$ найдется такое $\bar{v} \in V$, что $\varphi_j(x, \bar{v}) > 0$, $j = 1, \ldots, m$.

Пусть X — линейное нормированное пространство с нормой $\|\cdot\|$, $\sigma \leqslant 1$. Положим $\alpha(x_1, x_2) = L\|x_1 - x_2\|^\sigma$, $\beta(x_1, x_2) = P\|x_1 - x_2\|$, $\gamma(r) = Mr^\sigma$, $\omega = Rr$. Тогда $\rho(x_1, x_2) = (L + M(P/R)^\sigma)\|x_1 - x_2\|^\sigma$ и условие (2.3), таким образом, является условием Гёльдера с константой $L + M(P/R)^\sigma$ и показателем степени σ. При $\sigma = 1$ оно превращается в условие Липшица.

В заключение отметим, что результаты теоремы 2.1 можно распространить на задачу отыскания кратного максимина со связанными переменными [258].

§ 3. Многокритериальные задачи оптимизации

В теории принятия решений часто рассматриваются ситуации, в которых интересы лица, принимающего решения, состоят в стремлении к увеличению всех компонент некоторого векторного критерия $f(x) = (f_1(x), \ldots, f_k(x))$, где x принадлежит некоторому множеству $K \subset \mathbf{R}^n$. Разработано множество различных принципов принятия решений в ситуациях такого рода [30, 62, 63, 124, 152, 171, 203, 204, 235]. В этом параграфе нас будет интересовать один из наиболее традиционных принципов, связанный с выделением из всего множества $Y = \{y = f(x) \mid x \in K\}$ множества неулучшаемых или оптимальных по Парето векторов.

1. Постановка задачи. Вектор $y^* = f(x^*)$ называется *неулучшаемым* на множестве K, если не существует такого вектора $y \in Y$, что $y \succ y^*$, т. е. $y_i \geqslant y_i^*$, $i = 1, \ldots, k$, и хотя бы для одного i неравенство строгое. Говорят также, что y^* *не доминируется* векторами множества Y или *оптимален по Парето* в Y. Точка $x^* \in K$ называется при этом *эффективной точкой*. Множество всех неулучшаемых на K векторов обозначим через Y^* и будем называть *множеством Парето*.

Во многих задачах решение (или его первая стадия) состоит в отыскании множества эффективных точек (на второй стадии может потребоваться максимизировать на множестве эффективных точек некоторый дополнительный критерий). Типичным примером является задача автоматизации проектирования сложных технических систем [140—144].

Пусть $\varepsilon_1 > 0, \ldots, \varepsilon_k > 0$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_k)$. Наша цель состоит в том, чтобы выбрав надлежащим образом точки $x_1, \ldots, x_N \in K$ и вычислив в них значения вектор-функции f, выделить во множестве $Y_\varepsilon \stackrel{\text{def}}{=} \{f(x_1), \ldots, f(x_N)\}$ подмножество Y^*_ε, являющееся ε-аппроксимацией по Парето множества Y^*.

Мы называем множество Y' *ε-аппроксимацией по Парето* множества Y'', если векторы множества Y' попарно несравнимы (т. е. в Y' не существует таких двух векторов u, v, что $u > v$) и

$$\sup_{y'' \in Y''} \inf_{y' \in Y'} \max_{i=1,\ldots,k} \frac{y''_i - y'_i}{\varepsilon_i} \leqslant 1,$$

иными словами (в предположении, что нижняя грань достигается), для любого $y'' \in Y''$ найдется такой вектор $y' \in Y'$, что $y'_i \geqslant y''_i - \varepsilon_i$ при всех $i = 1, \ldots, k$.

При этом число точек N должно быть минимальным целым, при котором возможно гарантировать достижение поставленной цели.

Как видим, рассматриваемая в данном параграфе постановка близка к постановке задачи для общей модели вычислений гл. 1 в случае фиксированной точности ε. Отличие состоит, в частности, в том, что теперь ε — вектор.

Предположим, что для любых $x, x' \in K$

$$|f_i(x) - f_i(x')| \leqslant M_i \rho(x, x'), \quad i = 1, \ldots, k, \tag{3.1}$$

где $M_1, \ldots, M_k$ — известные константы, а ρ — функция, обладающая свойствами квазиметрики. Кроме того, для упрощения построений предположим, что

$$\varepsilon_1/M_1 = \ldots = \varepsilon_k/M_k \stackrel{\text{def}}{=} \delta. \tag{3.2}$$

Рассмотрим вопрос о построении оптимальных алгоритмов решения многокритериальных задач. Любой алгоритм решения поставленной задачи состоит, как и в общей модели вычислений гл. 1, из двух этапов.

На первом этапе, производя информационные вычисления (в нашем случае информационное вычисление — это вычисление значения вектор-функции f в точке множества K), получают информацию о задаче. Второй этап состоит в построении на основе полученной информации приближения к искомому решению (в нашем случае — выделении из множества Y_ε подмножества Y^*_ε). На

практике второй этап начинается не обязательно после завершения первого. В случае, если применяется последовательный алгоритм, промежуточные результаты второго этапа могут использоваться на первом этапе при выборе точек информационных вычислений.

В нашей задаче для всех рассматриваемых алгоритмов выделяемое на втором этапе из Y_ε подмножество Y_ε^* будет состоять из всех векторов, неулучшаемых на множестве $K_\varepsilon \overset{\text{def}}{=} \{x_1, \ldots, x_N\}$. Иными словами, Y_ε^* состоит из всех тех векторов множества Y_ε, которые не доминируются векторами множества Y_ε. Эффективная организация вычислений второго этапа не вызывает принципиальных затруднений, поэтому в дальнейшем под алгоритмом будем понимать лишь способ выбора точек $x_1, \ldots, x_N$.

Обозначим класс всех вектор-функций (задач) f, удовлетворяющих условиям (3.1), через F. Назовем алгоритм *оптимальным* в данном классе алгоритмов, если он решает любую задачу из F не более чем за N шагов (произведя не более N информационных вычислений) и не существует алгоритма, принадлежащего тому же классу, который решал бы любую задачу из F менее чем за N шагов. При этом под решением задачи мы понимаем построение ε-аппроксимации по Парето Y_ε^* множества Y^*.

2. Оптимальные алгоритмы решения многокритериальных задач оптимизации. Следующие два утверждения (и их доказательства) аналогичны соответствующим утверждениям для обычных оптимизационных задач (см. гл. 4). В формулировках предполагается, что число δ задано формулой (3.2) и что в K существует конечная δ-сеть, т. е. множество $A \subset K$, состоящее из конечного числа элементов и такое, что $\bigcup\limits_{a \in A} \{x \mid \rho(x, a) \leqslant \delta\} \supset K$.

Теорема 3.1 [253]. *Алгоритм, в качестве K_ε выбирающий δ-сеть с минимальным числом элементов, оптимален в классе пассивных алгоритмов.*

Доказательство. Обозначим через N_δ минимальное число элементов, из которых можно образовать δ-сеть в K. Очевидно, что векторы из множества Y_ε^* неулучшаемых на K_ε векторов попарно несравнимы. Пусть $y'' = f(x'') \in Y^*$, $x'' \in K$. В K_ε найдется такой элемент x, что $\rho(x, x'') \leqslant \delta$. Согласно (3.1), $|y_i - y_i''| \leqslant \varepsilon_i$, где $y = f(x)$, $i = 1, \ldots, k$. Если $y \notin Y_\varepsilon^*$, то найдется вектор

$y' \in Y_\varepsilon^*$ такой, что $y' > y$. Для него $y_i' > y_i \geqslant y_i'' - \varepsilon_i$, $i = 1, \ldots, k$. Тем самым показано, что рассматриваемый алгоритм решает любую задачу из F за N_δ шагов.

Пусть существует пассивный алгоритм $(x_1^0, \ldots x_\nu^0)$, решающий любую задачу из F за $\nu < N_\delta$ шагов. По определению числа N_δ существует такая точка $b \in K$, что $\rho(x_i^0, b) \geqslant \delta + \gamma$, $\quad i = 1, \ldots, \nu$, где $\gamma > 0$. Положим

$$f_j(x) = \max\{-M_j\rho(x, b) + M_j\delta + M_j\gamma, 0\}, \quad j = 1, \ldots, k.$$

В силу лемм 2.1, 2.7 гл. 1 $f \in F$. Нетрудно видеть, что

$$Y^* = \{(\varepsilon_1 + M_1\gamma, \ldots, \varepsilon_k + M_k\gamma)\},$$

$$f(x_i^0) = 0, \quad i = 1, \ldots, \nu, \quad Y_\varepsilon^* = \{(0, \ldots, 0)\}.$$

Очевидно, что Y_ε^* не является ε-аппроксимацией Y^*. Полученное противоречие завершает доказательство. △

Теорема 3.2 [253]. *Не существует последовательного алгоритма, который решал бы любую задачу из F менее чем за N_δ шагов. Таким образом, любой пассивный алгоритм оптимален в классе всех последовательных алгоритмов.*

Доказательство. Пусть существует последовательный алгоритм $\tilde{x}_0^\nu$, который решает любую задачу из F менее чем за N_δ шагов. Обозначим через $x_1^0, \ldots, x_\nu^0 (\nu < N_\delta)$ точки информационных вычислений при применении алгоритма $\tilde{x}_0^\nu$ к вектор-функции, все компоненты которой тождественно равны нулю. Ясно, что, применяя $\tilde{x}_0^\nu$ к вектор-функции f, построенной при доказательстве теоремы 3.1, получим те же самые точки информационных вычислений $x_1^0, \ldots, x_\nu^0$ и нулевые значения вектор-функции f в этих точках. Таким образом, в данном случае алгоритм $\tilde{x}_0^\nu$ не позволяет построить решение. Полученное противоречие завершает доказательство. △

Итак, применение последовательных алгоритмов не позволяет уменьшить число информационных вычислений при решении наихудшей задачи из F. Однако существуют алгоритмы, которые в благоприятных случаях позволяют решить задачу значительно быстрее.

Построение последовательно-оптимального по числу информационных вычислений алгоритма требует решения сложных вспомогательных задач, что делает нецелесообразным его практическое применение. Ограничимся поэтому построением одного из оптимальных алгоритмов,

который, не являясь последовательно-оптимальным, все же в благоприятных случаях значительно сокращает по сравнению с оптимальным пассивным алгоритмом число информационных вычислений и в то же время после некоторой модификации допускает достаточно простую программную реализацию. В силу определения оптимальности, в неблагоприятных случаях число информационных вычислений для обоих алгоритмов — одно и то же.

На первом шаге в качестве точки первого информационного вычисления выбирается произвольная точка δ-сети с N_δ элементами. В качестве x_2 выбирается любая другая точка той же сети.

Пусть сделано i шагов, $i \geqslant 2$, $x^i = (x_1, \ldots, x_i)$, $f(x_m) = y_m = (y_{m1}, \ldots, y_{mk})$, $m = 1, \ldots, i$, $y^i = (y_1, \ldots, y_i)$, $z^i = (x^i, y^i)$. Положим

$$\varphi_{2ij}(x) = \min_{m=1,\ldots,i} \{y_{mj} + M_j \rho(x, x_m)\}, \quad j = 1, \ldots, k.$$

Вектор-функция $\varphi_{2i} \overset{\text{def}}{=} (\varphi_{2i1}, \ldots, \varphi_{2ik})$ (ср. с обозначением (1.3) гл. 2 для скалярного случая) является, как следует из лемм 1.1, 1.2 гл. 2, точной мажорантой для вектор-функций из F, принимающих в точках $x_1, \ldots, x_i$ соответственно значения $y_1, \ldots, y_i$.

Положим

$$C_i = K \setminus \bigcup_{m=1}^{i} \{x \mid \varphi_{2i}(x) \leqslant y_m + \varepsilon\} =$$

$$= K \setminus \bigcup_{m=1}^{i} \bigcap_{j=1}^{k} \{x \mid \varphi_{2ij}(x) \leqslant y_{mj} + \varepsilon_j\} =$$

$$= K \setminus \bigcup_{m=1}^{i} \bigcap_{j=1}^{k} \bigcup_{p=1}^{k} \left\{x \mid \rho(x, x_p) \leqslant \frac{y_{mj} - y_{pj}}{M_j} + \delta\right\}. \quad (3.3)$$

Если $C_i = \varnothing$, то вычисления прекращаются. Если $C_i \neq \varnothing$, то обозначим через $N_\delta(z^i)$ минимальное число элементов, принадлежащих K (но не обязательно C_i), из которых можно образовать δ-сеть для C_i. На $(i+1)$-м шаге в качестве x_{i+1} выбирается произвольная точка δ-сети для C_i с $N_\delta(z^i)$ элементами.

Определенный таким образом алгоритм обозначим через α.

Теорема 3.3 [253]. *Алгоритм α оптимален.*

Доказательство. Пусть $\{x^i_{i+1}, \ldots, x^i_{i+N_\delta(z^i)}\}$ — δ-сеть для C_i и пусть $y'' = f(x'') \in Y^*$, $x'' \in K$. Если $x'' \in C_i$,

то найдется такая точка $x \in \left\{x^i_{i+1}, \ldots, x^i_{i+N\delta(z^i)}\right\}$, что $\rho(x, x'') \leqslant \delta$, и, следовательно, $|y_j - y''_j| \leqslant \varepsilon_j$, где $y = f(x)$, $j = 1, \ldots, k$. Если же $x'' \notin C_i$, то, по определению C_i, найдется точка $x \in \{x_1, \ldots, x_i\}$ такая, что $\varphi_{2i}(x'') \leqslant f(x) + \varepsilon$. В этом случае $y'' \leqslant y + \varepsilon$, где $y = f(x)$, так как φ_{2i} — мажоранта. Полученные неравенства доказывают, что выполнение оставшихся вычислений в точках $x^i_{i+1}, \ldots, x^i_{i+N_\delta(z^i)}$ гарантирует отыскание решения задачи.

Из (3.3) выводим

$$C_i \subset C_{i-1} \setminus \{x \mid \rho(x, x_i) \leqslant \delta\}. \tag{3.4}$$

Отсюда следует, что

$$N_\delta(z^i) \leqslant N_\delta(z^{i-1}) - 1, \tag{3.5}$$

так как, по построению, x_i — одна из точек δ-сети для C_{i-1} с $N_\delta(z^{i-1})$ элементами, а остальные $N_\delta(z^{i-1}) - 1$ точек этой сети образуют, в силу (3.4), δ-сеть для C_i. Неравенство (3.5) справедливо при любом $i \geqslant 2$, кроме того, $N_\delta(z^1) \leqslant N_\delta - 1$, т. е.

$$N_\delta(z^i) \leqslant N_\delta(z^{i-1}) - 1 \leqslant N_\delta(z^{i-2}) - 2 \leqslant \ldots \leqslant N_\delta - i.$$

Таким образом, для любого i точка x_{i+1} выбирается так, что отыскание решения гарантируется не более чем за $N_\delta - i$ шагов. Итак, общее число шагов не превышает N_δ и, следовательно, алгоритм α оптимален. $\triangle$

Отметим, что алгоритм α является аналогом алгоритма из п. 3 § 3 гл. 4.

3. О программной реализации оптимального алгоритма. Непосредственная программная реализация алгоритма α для случая произвольных K и ρ практически невозможна из-за необходимости построения на каждом шаге δ-сетей с минимальным числом элементов, что требует, в свою очередь, решения сложных геометрических задач построения оптимальных покрытий. Сделаем поэтому ряд упрощающих предположений.

Пусть K является n-мерным координатным параллелепипедом с ребрами длины $2\delta m_1, \ldots, 2\delta m_n$, где $m_1, \ldots, m_n$ — целые числа, $\rho(u, v) = \max\limits_{i=1,\ldots,n} |u^i - v^i|$. Разобьем ребра K соответственно на $m_1, \ldots, m_n$ равных частей и проведем через точки разбиения гиперплоскости, параллельные координатным гиперплоскостям. При этом K разобьется на $N = m_1 \cdot \ldots \cdot m_n$ координатных кубов с реб-

рами длины 2δ. Обозначим центры кубов через $u_1, \ldots, u_N$. Ясно, что множество $U \overset{\text{def}}{=} \{u_1, \ldots, u_N\}$ является для K δ-сетью с минимальным числом элементов и, следовательно, $N = N_\delta$.

Основное упрощение исходного алгоритма α основано на той же идее, что и алгоритм (4.14), (4.15) гл. 4, и состоит в том, что после i шагов в качестве δ-сети для множества K выбирается δ-сеть, содержащая только точки множества U. Такой δ-сетью, как нетрудно видеть, является множество

$$U_i \overset{\text{def}}{=} U \setminus \bigcup_{m=1}^{i} \{u \mid \varphi_{2i}(u) \leqslant y_m\}. \tag{3.6}$$

Очевидно, что в (3.6) достаточно брать объединение только тех множеств, которые соответствуют векторам y_m, не доминируемым векторами y_j, $j = 1, \ldots, m, m+1, \ldots$ $\ldots, i$. Принятые допущения позволяют построить достаточно простую программную реализацию алгоритма многокритериальной оптимизации, хотя при этом число элементов множества U_i может оказаться больше $N_\delta(z^i)$.

Для организации работы алгоритма в памяти выделяется массив из N элементов. Элемент, соответствующий j-й точке (точке u_j), находится на j-м месте в массиве (необходимо иметь программу, устанавливающую взаимно однозначное соответствие между номерами и координатами точек) и содержит после i шагов следующие записи: k компонент вектор-функции f для элементов определяемого ниже списка 1, k компонент вектор-функции φ_{2i} для элементов списка 2, а также необходимые указатели.

На первом шаге случайно равновероятно выбирается $j_1 \in \{1, \ldots, N\}$, принимается $x_1 = u_{j_1}$, вычисляется $y_1 = f(x_1)$ и организуются два связных списка. В список 1 включается элемент, соответствующий точке x_1, в список 2 — остальные элементы. Для элементов списка 2 вычисляются значения φ_{2i}.

Пусть сделано i шагов, вычислены векторы $y_1, \ldots, y_i$. В списке 1 содержатся элементы, соответствующие тем из вычисленных векторов, которые не доминируются векторами множества $\{y_1, \ldots, y_i\}$. Элементы, соответствующие доминируемым векторам, находятся в числе исключенных элементов. Список 2 состоит из неисключенных элементов.

На $(i+1)$-м шаге случайно равновероятно выбирается элемент из списка 2. Пусть выбран j-й элемент массива. Вектор $\varphi_{2i}(u_j)$ сравнивается поочередно с векторами y_m списка 1.

Если $\varphi_{2i}(u_j) \leqslant y_m$ для какого-то из них, то выбирается элемент, соседний с j-м в списке 2, сам j-й элемент массива исключается, а вектор φ_{2i}, соответствующий вновь выбранному элементу, сравнивается поочередно с векторами y_m списка 1.

В противном случае принимается $x_{i+1} = u_j$, вычисляются y_{i+1},

$$\varphi_{2,i+1,p}(u_l) = \min\{\varphi_{2ip}(u_l),\ y_{i+1,p} + M_p\rho(x_{i+1},\ u_l)\},$$
$$p = 1, \ldots, k,$$

для всех точек u_l, относящихся к элементам списка 2. Вектор y_{i+1} сравнивается поочередно с векторами y_m списка 1. Если $y_{i+1} \leqslant y_m$ для какого-либо из них, то j-й элемент массива (он соответствует вектору y_{i+1}) исключается. В противном случае из списка 1 исключаются векторы, доминируемые вектором y_{i+1} (если таковые найдутся), а j-й элемент включается в список 1.

Вычисления прекращаются, если список 2 пуст. Векторы y_m списка 1 дают искомое решение.

В заключение несколько замечаний по поводу возможных модификаций исходной постановки задачи и алгоритма α.

На практике константы $M_1, \ldots, M_k$ часто неизвестны или известны неточно. В этом случае алгоритм должен начинаться с блока оценки констант. Такой блок можно построить, исходя из рекомендаций п. 2 § 4 гл. 2. Конечно, алгоритмы с блоком оценки констант уже не гарантируют отыскание решения задачи и, тем более, не обладают свойством оптимальности, но, как нетрудно показать, в данных условиях гарантировать построение решения за N шагов невозможно ни при каком N.

В некоторых ситуациях бывает необходимо приближенно найти не все множество неулучшаемых векторов Y^*, а какое-то его подмножество, например, $Y^*(w) = \{y \mid y \in Y^*,\ y_i \geqslant w_i,\ i = 1, \ldots, k\}$. В этом случае на $(i+1)$-м шаге, выбрав из списка 2 j-й элемент массива, следует сравнить вектор $\varphi_{2i}(u_j)$ сначала с вектором $w - \varepsilon$. Если $\varphi_{2is}(u_j) < w_s - \varepsilon_s$ хотя бы для одного s, то j-й элемент исключается. Необходимо произвести также и другие очевидные изменения в алгоритме.

Отметим, что описанный в данном пункте алгоритм, несмотря на его очевидные недостатки по сравнению с оптимальным пассивным алгоритмом, т. е. методом полного перебора элементов множества U (бо́льшая память, необходимость организации связных списков, вычисления векторов φ_{2i}, сравнения их с векторами f и т. д.), способен при высокой сложности информационных вычислений принести значительную экономию машинного времени.

Заметим, наконец, что задача аппроксимации множества Парето является неустойчивой относительно погрешностей вычисления частных критериев $f_1, \ldots, f_k$ и может быть регуляризована методами, развиваемыми в [175, 176, 210, 235, 271, 286].

СПИСОК ЛИТЕРАТУРЫ

1. Акимова И. Я. Применение диаграмм Вороного в комбинаторных задачах: Обзор // Изв. АН СССР. Сер. техн. киберн.— 1984.— № 2.— С. 102—109.
2. Альперович Э. Е., Батищев Д. И., Стропгин Р. Г. Теоретические и прикладные аспекты тестирования алгоритмов поиска // Вопросы кибернетики. Проблемы случайного поиска.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1973.— С. 53—56.
3. Анучина Н. Н., Бабенко К. И., Годунов С. К. и др. Теоретические основы конструирования численных алгоритмов задач математической физики/Под ред. К. И. Бабенко.— М.: Наука, 1979.— 296 с.
4. Афанасьев А. Ю. О поиске минимума функции с ограниченной второй производной // ЖВМиМФ.— 1974.— Т. 14, № 4. С. 1018—1021.
5. Афанасьев А. Ю., Новиков В. А. О поиске минимума функции с ограниченной третьей производной // ЖВМиМФ.— 1977.— Т. 17, № 4.— С. 1031—1034.
6. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов.— М.: Мир, 1979.— 536 с.
7. Бабенко В. Ф. Асимптотически точная оценка остатка наилучших для некоторых классов функций кубатурных формул // Мат. заметки.— 1976.— Т. 19, вып. 3.— С. 313—322.
8. Бабенко В. Ф. Точная асимптотика остатков оптимальных для некоторых классов функций весовых кубатурных формул // Мат. заметки.— 1976.— Т. 20, вып. 4.— С. 589—595.
9. Бабий А. Н. Алгоритм нахождения значения глобального экстремума функции нескольких переменных с заданной точностью // Кибернетика.— 1978.— № 5.— С. 52—56.
10. Бабич М. Д., Шевчук Л. Б. Об одном алгоритме приближенного решения систем нелинейных уравнений // Кибернетика.— 1982.— № 2.— С. 74—79.
11. Батищев Д. И. Поисковые методы оптимального проектирования.— М.: Сов. радио, 1975.— 216 с.
12. Батухтин В. Д., Майборода Л. А. Методы поиска экстремума разрывных функций // Изв. АН СССР. Сер. техн. киберн.— 1984.— № 4.— С. 192—202.
13. Бахвалов Н. С. О приближенном вычислении кратных интегралов // Вестн. Моск. ун-та. Сер. мат., механ., астрон., физ., хим.— 1959.— № 4.— С. 3—18.
14. Бахвалов Н. С. Оценка в среднем остаточного члена квадратурных формул // ЖВМиМФ.— 1961.— Т. 1, № 1.— С. 64—77.

15. Бахвалов Н. С. Об определении начального шага и оценке главного члена погрешности при численном интегрировании с автоматическим выбором шага // Вычислительные методы и программирование, № 1.— М.: Изд-во Моск. ун-та, 1962.— С. 69—79.
16. Бахвалов Н. С. Об оптимальных оценках скорости сходимости квадратурных процессов и методов интегрирования типа Монте-Карло на классах функций // Численные методы решения дифференциальных и интегральных уравнений и квадратурные формулы.— М.: Наука, 1964.— С. 5—63.
17. Бахвалов Н. С. О вычислении кратных интегралов с автоматическим выбором шага // Вычислительные методы и программирование, № 3.— М.: Изд-во Моск. ун-та, 1965.— С. 237—240.
18. Бахвалов Н. С. Об алгоритмах выбора шага интегрирования // Вычислительные методы и программирование, № 5.— М.: Изд-во Моск. ун-та, 1966.— С. 3—8.
19. Бахвалов Н. С. О свойствах оптимальных методов решения задач математической физики // ЖВМиМФ.— 1970.— Т. 10, № 3.— С. 555—568.
20. Бахвалов Н. С. Об оптимальности линейных методов приближения операторов на выпуклых классах функций // ЖВМиМФ.— Т. 11, № 4.— С. 1014—1018.
21. Бахвалов Н. С. Оценки снизу асимптотических характеристик классов функций с доминирующей смешанной производной // Мат. заметки.— 1972.— Т. 12, № 6.— С. 655—664.
22. Бахвалов Н. С. Численные методы.— М.: Наука, 1973.— 632 с.
23. Белая Н. И. Алгоритм построения оптимальной по точности производной функции в классе $C_{2,L,N}$ // Изв. вузов. Сер. Математика.— 1978.— № 8.— С. 31—43.
24. Белая Н. И., Иванов В. В. Об оптимальных по точности алгоритмах восстановления производных функций некоторых классов // ЖВМиМФ.— 1985.— Т. 25, № 3.— С. 456—461.
25. Беллман Р. Динамическое программирование.— М.: ИЛ, 1960.— 400 с.
26. Беллман Р., Дрейфус С. Прикладные задачи динамического программирования.— М.: Наука, 1965.— 460 с.
27. Березин И. С., Жидков Н. П. Методы вычислений. Т. I.— М.: Наука, 1966.— 632 с.
28. Березовский А. И., Данильченко Л. С., Дульская В. А., Иванов В. В. Аппроксимация функций.— Препринт/Ин-т киберн.— Киев.— 1979.— 79—48.— 40 с.
29. Березовский А. И., Иванов В. В. Об оптимальной по точности равномерной сплайновой аппроксимации // Изв. вузов. Сер. Математика.— 1977.— № 10.— С. 14—24.
30. Березовский Б. А., Борзенко В. И., Кемпер Л. М. Бинарные отношения в многокритериальной оптимизации.— М.: Наука, 1981.— 150 с.
31. Бойков И. В. Оптимальные по точности алгоритмы приближенного вычисления сингулярных интегралов.— Саратов: Изд-во Сарат. ун-та, 1983.— 210 с.
32. Бойков И. В. Оптимальные методы вычислений в задачах автоматического регулирования.— Пенза: Изд-во Пенз. политехн. ин-та, 1983.— 96 с.

33. Боянов Б. Д. Оптимальная скорость интегрирования и ε-энтропия одного класса аналитических функций // Мат. заметки.— 1973.— Т. 14, № 1.— С. 3—10.
34. Боянов Б. Д. Наилучшие методы интерполирования для некоторых классов дифференцируемых функций // Мат. заметки.— 1975.— Т. 17, № 4.— С. 511—524.
35. Брусов В. С., Пиявский С. А. Вычислительный алгоритм оптимального покрытия областей плоскости // ЖВМиМФ.— 1971.— Т. 11, № 2.— С. 304—312.
36. Вальд А. Последовательный анализ.— М.: Физматгиз, 1960.— 328 с.
37. Вальд А. Статистические решающие функции. // Позиционные игры.— М.: Наука, 1967.— С. 300—522.
38. Васильев Н. С. К отысканию глобального минимума квазивогнутой функции // ЖВМиМФ.— 1983.— Т. 23, № 2.— С. 307—313.
39. Васильев Н. С. Активный вычислительный метод поиска глобального минимума вогнутой функции // ЖВМиМФ.— 1984.— Т. 24, № 1.— С. 152—156.
40. Васильев Н. С. Поиск минимума в вогнутых задачах с использованием достаточного условия глобального экстремума // ЖВМиМФ.— 1985.— Т. 25, № 2.— С. 190—199.
41. Васильев Н. С. Аппроксимация выпуклых множеств многогранниками и ее приложения к поиску глобального экстремума // Техн. киберн.— 1986.— № 1.— С. 27—36.
42. Васильев П. П. Оптимальный поиск нуля функции, вычисляемой приближенно // Математические методы оптимизации и управления в сложных системах.— Калинин: Изд-во КГУ, 1983.— С. 19—25.
43. Васильев П. П. Оптимальный поиск корня функции, вычисляемой приближенно.— Деп. в ВИНИТИ 28.06.83, № 3486—83.— 13 с.
44. Васильев П. П. Об оптимальном на один шаг стохастическом поиске корня функции, удовлетворяющей условию Липшица.— Деп. в ВИНИТИ 10.02.84, № 827—84.— 37 с.
45. Васильев П. П. Об оптимальном пассивном поиске корня монотонной функции // Вопросы кибернетики. Модели и методы глобальной оптимизации.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1985.— С. 98—111.
46. Васильев П. П., Сухарев А. Г. Программа численного решения системы нелинейных уравнений методом Брауна с предварительным подбором начального приближения.— ГосФАП, № П006441, 1983 (Аннотация в сб.: Алгоритмы и программы.— 1983.— № 5.— С. 20).
47. Васильев С. Б., Ганшин Г. С. Последовательный алгоритм поиска наибольшего значения дважды дифференцируемой функции // Мат. заметки.— 1982.— Т. 31, № 4.— С. 613—618.
48. Васильев Ф. П. Численные методы решения экстремальных задач.— М.: Наука, 1980.— 520 с.
49. Васильев Ф. П. Методы решения экстремальных задач.— М.: Наука, 1981.— 400 с.
50. Васильева Л. Г. и др. Процедуры численного интегрирования на языке АЛГОЛ.— М.: Изд-во Моск. ун-та, 1972.— 81 с.
51. Винер Н. Интеграл Фурье и некоторые его применения.— М.: Физматгиз, 1963.— 256 с.

52. Витушкин А. Г. Оценка сложности задач табулирования.— М.: Физматгиз, 1959.— 228 с.
53. Волков Е. А. О поиске максимума функции и приближенном глобальном решении систем нелинейных уравнений // Тр. МИАН СССР.— 1974.— Т. 131.— С. 64—80.
54. Высоцкая И. Н., Стронгин Р. Г. Метод решения нелинейных уравнений, использующий априорные вероятностные оценки корней // ЖВМиМФ.— 1983.— Т. 23, № 1.— С. 3—12.
55. Габдулхаев Б. Г. Оптимальные аппроксимации решений линейных задач.— Казань: Изд-во Каз. ун-та, 1980.— 232 с.
56. Гамецкий А. Ф. Об оптимальности главной решетки первого типа Вороного среди решеток первого типа любого числа измерений // ДАН СССР.— 1963.— Т. 151, № 3.— С. 482—484.
57. Ганшин Г. С. Вычисление наибольшего значения функций // ЖВМиМФ.— 1976.— Т. 16, № 1.— С. 30—39.
58. Ганшин Г. С. Оптимальные пассивные алгоритмы вычисления наибольшего значения функций на отрезке // ЖВМиМФ.— 1977.— Т. 17, № 3.— С. 562—571.
59. Ганшин Г. С. Вычисление наибольшего значения функций нескольких переменных // Кибернетика.— 1983.— № 2.— С. 61—63.
60. Гапоненко Ю. Л. Об одном методе отыскания глобального экстремума для нелинейного функционала // Кибернетика.— 1981.— № 6.— С. 58—61.
61. Гермейер Ю. Б. Методологические и математические основы исследования операций и теории игр.— М.: Изд-во Моск. ун-та, 1967.
62. Гермейер Ю. Б. Введение в теорию исследования операций.— М.: Наука, 1971.— 384 с.
63. Гермейер Ю. Б. Игры с непротивоположными интересами.— М.: Наука, 1976.— 328 с.
64. Гермейер Ю. Б., Крылов И. А. Поиск максиминов методом «невязок» // ЖВМиМФ.— 1972.— Т. 12, № 4.— С. 871—881.
65. Гирлин С. К. Об оптимальных по точности интерполяции и минимизации функций класса $C_{2,L_1,L_2,\ldots,L_m,N_2}$ // Изв. вузов. Сер. Математика.— 1978.— № 10.— С. 95—98.
66. Гирлин С. К., Ганшин Г. С. Оптимальные алгоритмы поиска наименьшего значения функции // Изв. вузов. Сер. Математика.— 1983.— № 3.— С. 75—77.
67. Глинкин И. А. Об оптимальном интегрировании монотонных функций // Математические методы в исследовании операций.— М.: Изд-во Моск. ун-та, 1981.— С. 37—46.
68. Глинкин И. А. Об оптимальном поиске экстремумов нескольких функций // Математические методы в исследовании операций.— М.: Изд-во Моск. ун-та, 1981.— С. 46—54.
69. Глинкин И. А. Об оптимальных алгоритмах интегрирования выпуклых функций // ЖВМиМФ.— 1983.— Т. 23, № 2.— С. 267—277.
70. Глинкин И. А. О наилучшей квадратурной формуле на классе выпуклых функций // Мат. заметки.— 1984.— Т. 35, № 5.— С. 697—707.
71. Глинкин И. А. О поиске глобального экстремума функций, представимых в виде интеграла по области переменной конфигурации // Вопросы кибернетики. Модели и методы глобаль-

ной оптимизации.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1985.— С. 91—97.

72. Глинкин И. А., Сухарев А. Г. Последовательно-оптимальный алгоритм вычисления интегралов.— ГосФАП, № П004578, 1980 (Аннотация в сб.: Алгоритмы и программы.— 1980.— № 6, П004578).
73. Глинкин И. А., Сухарев А. Г. Вычисление кратных интегралов с помощью разверток типа кривой Пеано // ЖВМиМФ.— 1984.— Т. 24, № 8.— С. 1259—1263.
74. Глинкин И. А., Сухарев А. Г. Исследование эффективности некоторых алгоритмов численного интегрирования и их применение для решения экстремальных задач // Вопросы кибернетики. Модели и методы глобальной оптимизации.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1985.— С. 23—37.
75. Глушков В. М. О диалоговом методе решения оптимизационных задач // Кибернетика.— 1975.— № 4.— С. 2—6.
76. Глушков В. М. О системной оптимизации // Кибернетика.— 1980.— № 5.— С. 89—90.
77. Глушков В. М., Иванов В. В., Михалевич В. С., Сергиенко И. В., Стогний А. А. Резервы оптимизации вычислений.— Препринт/Ин-т киберн.— Киев, 1977.— 77—67.— 54 с.
78. Глушков В. М., Иванов В. В., Яненко В. М. Моделирование развивающихся систем.— М.: Наука, 1983.— 352 с.
79. Гончарский А. В., Леонов А. С., Ягола А. Г. Обобщенный принцип невязки // ЖВМиМФ.— 1973.— Т. 13, № 2.— С. 294—302.
80. Городецкий С. Н., Неймарк Ю. И., Фадеев В. А. Непараметрические адаптивные стохастические модели в задачах поисковой глобальной оптимизации // Математическая статистика и ее приложения, вып. 9.— Томск: Изд-во ТГУ, 1983.— С. 52—58.
81. Гребенников А. И. Об оптимальном приближении нелинейных операторов // ЖВМиМФ.— 1978.— Т. 18, № 3.— С. 762—766.
82. Гребенников А. И. Метод сплайнов и решение некорректных задач теории приближений.— М.: Изд-во Моск. ун-та, 1983.— 208 с.
83. Гришагин В. А., Стронгин Р. Г. Оптимизация многоэкстремальных функций при монотонно унимодальных ограничениях // Изв. АН СССР. Сер. техн. киберн.— 1984.— № 4.— С. 203—208.
84. Громенко В. М., Гурин Л. С. Об оптимальном алгоритме поиска экстремума для полиномов, ограниченных по модулю // Автоматика и вычисл. техн.— 1974.— № 2.— С. 49—56.
85. Гупал А. М. Стохастические методы решения негладких экстремальных задач.— Киев: Наук. думка, 1979.— 152 с.
86. Гурин Л. С., Дымарский Я. С., Меркулов А. Д. Задачи и методы оптимального распределения ресурсов.— М.: Сов. радио, 1968.— 464 с.
87. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи.— М.: Мир, 1982.— 416 с.
88. Данилин Ю. М. Оценка эффективности одного алгоритма

отыскания минимума // ЖВМиМФ.— 1971.— Т. 11, № 4.— С. 1026—1031.
89. Данилин Ю. М., Пиявский С. А. Об одном алгоритме отыскания абсолютного минимума // Семинар. Теория оптимальных решений, вып. 2.— Киев: Изд-во ИК АН УССР, 1967.— С. 25—37.
90. Демьянов В. Ф., Малоземов В. Н. Введение в минимакс.— М.: Наука, 1972.— 368 с.
91. Демьянов В. Ф., Васильев Л. В. Недифференцируемая оптимизация.— М.: Наука, 1981.— 384 с.
92. Евтушенко Ю. Г. Численный метод поиска глобального экстремума // ЖВМиМФ.— 1971.— Т. 11, № 6.— С. 1390—1403.
93. Евтушенко Ю. Г. Методы поиска глобального экстремума // Исследование операций.— М.: ВЦ АН СССР, 1974.— С. 39—68.
94. Евтушенко Ю. Г. Методы решения экстремальных задач и их применение в системах оптимизации.— М.: Наука, 1982.— 432 с.
95. Ермаков С. М. Метод Монте-Карло и смежные вопросы.— М.: Наука, 1971.— 328 с.
96. Ермольев Ю. М. Методы стохастического программирования.— М.: Наука, 1978.— 240 с.
97. Женсыкбаев А. А. Моносплайны минимальной нормы и наилучшие квадратурные формулы // УМН.— 1981.— Т. 36, вып. 4 (220).— С. 107—159.
98. Жидков Н. П. Линейные аппроксимации функционалов.— М.: Изд-во Моск. ун-та, 1977.— 262 с.
99. Жилейкин Я. М., Кукаркин А. Б. Об оптимальном вычислении интегралов от быстроосциллирующих функций // ЖВМиМФ.— 1978.— Т. 18, № 2.— С. 294—301.
100. Жилинскас А. Г. Одношаговый байесовский метод поиска экстремума функции одной переменной // Кибернетика.— 1975.— № 1.— С. 139—144.
101. Жилинскас А. Г. Метод одномерной многоэкстремальной минимизации // Изв. АН СССР. Сер. техн. киберн.— 1976.— № 4.— С. 71—74.
102. Жилинскас А. Г. Глобальная оптимизация. Аксиоматика статистических моделей, алгоритмы, применения.— Вильнюс: Мокслас, 1986.— 168 с.
103. Жиглявский А. А. Математическая теория глобального случайного поиска.— Л.: Изд-во Ленингр. ун-та, 1985.— 292 с.
104. Жиров В. С. Поиск глобального экстремума многочлена на параллелепипеде // ЖВМиМФ.— 1985.— Т. 25, № 2.— С. 163—180.
105. Загускин В. Л. Справочник по численным методам решения алгебраических и трансцендентных уравнений.— М.: Физматгиз, 1960.— 216 с.
106. Задирака В. К. Теория вычисления преобразования Фурье.— Киев: Наук. думка, 1983.— 215 с.
107. Задирака В. К., Иванов В. В. Вопросы оптимизации вычислений (численное интегрирование и дифференцирование).— Киев: Об-во «Знание» УССР, 1979.— 44 с.
108. Задирака В. К., Игисинов К. Анализ точности и эффективности алгоритмов быстрого преобразования Фурье и

некоторых его приложений // Кибернетика.— 1973.— № 6.— С. 57—61.
109. Зализняк Н. Ф., Лигун А. А. Об оптимальных стратегиях поиска глобального максимума функции // ЖВМиМФ.— 1978.— Т. 18, № 2.— С. 314—321.
110. Иванов В. В. Об оптимальных алгоритмах численного решения сингулярных интегральных уравнений // Механика сплошной среды и родственные проблемы анализа.— М.: Наука, 1972.— С. 209—219.
111. Иванов В. В. Об оптимальных алгоритмах вычисления сингулярных интегралов // ДАН СССР.— 1972.— Т. 204, № 1.— С. 21—24.
112. Иванов В. В. Об оптимальных алгоритмах минимизации функций некоторых классов // Кибернетика.— 1972.— № 4.— С. 81—94.
113. Иванов В. В. Об оптимальных по точности алгоритмах приближенного решения операторных уравнений 1-го рода // ЖВМиМФ.— 1975.— Т. 15, № 1.— С. 3—11.
114. Иванов В. В. Об оптимальных по точности алгоритмах приближения функций некоторых классов // Теория приближения функций.— М.: Наука, 1977.— С. 195—200.
115. Иванов В. В. О минимизации числа операций для линейных некорректных задач // Изв. вузов. Сер. Математика.— 1978.— № 11.— С. 47—54.
116. Иванов В. В. Методы вычислений на ЭВМ. Справочное пособие.— Киев: Наук. думка, 1986.— 584 с.
117. Иванов В. В., Задирака В. К. Вопросы оптимизации вычислений.— Киев: Об-во «Знание» УССР, 1979.— 36 с.
118. Иванов В. В., Коржова В. Н. Об оптимизации алгоритмов оценок некоторых вероятностных характеристик // Кибернетика.— 1983.— № 3.— С. 94—102.
119. Иванов В. В., Людвиченко В. А., Михалевич В. С., Трубин В. А., Шор Н. З. Вопросы повышения эффективности алгоритмов минимизации функций и математического программирования.— Препринт/Ин-т киберн.— Киев, 1979.— 79—59.— 54 с.
120. Иванов В. К., Васин В. В., Танана В. П. Теория линейных некорректных задач и ее приложения.— М.: Наука, 1978.— 206 с.
121. Калинин И. Н. К вопросу исследования и сравнения алгоритмов оптимизации // Кибернетика.— 1984.— № 1.— С. 77—80.
122. Карлин С. Математические методы в теории игр, программировании и экономике.— М.: Мир, 1964.— 840 с.
123. Карманов В. Г. Математическое программирование.— М.: Наука, 1986.— 288 с.
124. Кини Р., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения.— М.: Радио и связь, 1981.— 560 с.
125. Коллатц Л. Функциональный анализ и вычислительная математика.— М.: Мир, 1969.— 448 с.
126. Колмогоров А. Н., Тихомиров В. М. ε-энтропия и ε-емкость множеств в функциональных пространствах // УМН.— 1959.— Т. 14, вып. 2 (86).— С. 3—80.
127. Колмогоров А. Н., Фомин С. В. Элементы теории функций и функционального анализа.— М.: Наука, 1976.— 496 с.

128. Конопов В. А., Бирюкова Т. Л. Об одном методе поиска экстремумов некоторого класса функций одной переменной // Вычислительная математика.— Киев: Ин-т киберн., 1975.— С. 89—98.
129. Корнейчук Н. П. Наилучшие кубатурные формулы для некоторых классов функций многих переменных // Мат. заметки.— 1968.— Т. 3, вып. 5.— С. 565—576.
130. Корнейчук Н. П. Экстремальные задачи теории приближений.— М.: Наука, 1976.— 320 с.
131. Корнейчук Н. П. Сплайны в теории приближения.— М.: Наука, 1984.— 352 с.
132. Корнейчук Н. П. Оптимальные методы кодирования и восстановления функций // Оптимални алгоритми.— София: БАН, 1986.— С. 157—171.
133. Коротченко А. Г. Об одном алгоритме поиска наибольшего значения одномерных функций // ЖВМиМФ.— 1978.— Т. 18, № 3.— С. 563—573.
134. Коротченко А. Г. О поиске минимумов нескольких унимодальных функций // ЖВМиМФ.— 1979.— Т. 19, № 5.— С. 1337—1340.
135. Корчанов С. В. Об оптимальных методах интегрирования унимодальных функций // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1984.— № 3.— С. 38—43.
136. Корчанов С. В. Об оптимальных алгоритмах интегрирования функций, имеющих известное количество экстремумов // Программное обеспечение и модели исследования операций.— М.: Изд-во Моск. ун-та, 1986.— С. 177—185.
137. Корчанов С. В. Оптимальные на один шаг алгоритмы интегрирования дважды дифференцируемых функций.— Деп. ВИНИТИ 02.10.86, № 6970—В86.— 23 с.
138. Корчанов С. В. Об оптимальных в классах произведений сеток алгоритмах восстановления повторного функционала // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1986.— № 4.— С. 25—30.
139. Корчанов С. В. Об оптимальных алгоритмах восстановления повторного функционала.— Деп. ВИНИТИ 08.04.86, № 2489—В86.— 26 с.
140. Краснощёков П. С., Математические модели в исследовании операций.— М.: Знание, 1984.— 64 с.
141. Краснощёков П. С., Морозов В. В., Фёдоров В. В. Декомпозиция в задачах проектирования // Изв. АН СССР. Сер. техн. киберн.— 1979.— № 2.— С. 7—17.
142. Краснощёков П. С., Морозов В. В., Фёдоров В. В. Последовательное агрегирование в задачах внутреннего проектирования технических систем // Изв. АН СССР. Сер. техн. киберн.— 1979.— № 5.— С. 5—12.
143. Краснощёков П. С., Петров А. А. Принципы построения моделей.— М.: Изд-во Моск. ун-та, 1983.— 264 с.
144. Краснощёков П. С., Петров А. А., Фёдоров В. В. Информатика и проектирование.— М.: Знание, 1986.— 48 с.
145. Кротов В. Ф., Пиявский С. А. Достаточные условия оптимальности в задачах об оптимальных покрытиях // Изв. АН СССР. Сер. техн. киберн.— 1968.— № 2.— С. 10—17.
146. Кругер А. Я. Об оптимальном поиске корня функции // Вестн. Белор. ун-та. Сер. I.— 1975.— № 3.— С. 3—4.

147. Кругер А. Я. К задаче поиска корня функции // Вестн. Белор. ун-та. Сер. I.— 1976.— № 1.— С. 3—6.
148. Кузовкин А. И., Тихомиров В. М. О количестве вычислений для нахождения минимума выпуклых функций // Эконом. и мат. методы.— 1967.— Т. 3, № 1.— С. 95—103.
149. Куп Г. Позиционные игры и проблема информации // Позиционные игры.— М.: Наука, 1967.— С. 13—40.
150. Куржанский А. Б. Управление и наблюдение в условиях неопределенности.— М.: Наука, 1977.— 392 с.
151. Курицкий А. М. Энергетика механических балансовых спусковых регуляторов // Тр. НИИЧаспрома, вып. 8.— М.: Изд-во НИИЧаспрома, 1971.— С. 66—81.
152. Ларичев О. И., Никифоров А. Д. Аналитический обзор процедур решения многокритериальных задач математического программирования // Экономика и мат. методы.— 1986.— Т. 22, № 3.— С. 508—523.
153. Лбов Г. С., Грунов А. А. Об одном алгоритме поиска глобального экстремума функции // Вычислительные системы, вып. 67.— Новосибирск: Наука, 1976.— С. 69—76.
154. Левин А. Ю. Об одном алгоритме минимизации выпуклых функций // ДАН СССР.— 1965.— Т. 160, № 6.— С. 1244—1247.
155. Леонов В. В. Метод покрытий для отыскания глобального максимума функций от многих переменных // Исследования по кибернетике.— М.: Сов. радио, 1970.— С. 41—52.
156. Лузин Н. Н. Теория функций действительного переменного.— М.: Учпедгиз, 1948.— 318 с.
157. Льюс Р., Райфа Х. Игры и решения.— М.: ИЛ, 1961.— 642 с.
158. Майстровский Г. Д. Об оптимальности метода Ньютона // ДАН СССР.— 1972.— Т. 204, № 6.— С. 1313—1315.
159. Маркин Д. Л., Стронгин Р. Г. Метод решения многоэкстремальных задач с невыпуклыми ограничениями, использующий априорную информацию об оценках оптимума // ЖВМиМФ.— 1987.— Т. 27, № 1.— С. 52—62.
160. Марчук А. Г. Оптимальные по точности методы решения линейных задач восстановления.— Препринт/ВЦ СО АН СССР.— Новосибирск, 1976.—29 с.
161. Марчук А. Г., Осипенко К. Ю. Наилучшее приближение функций, заданных с погрешностью в конечном числе точек // Мат. заметки.— 1975.— Т. 17, № 3.— С. 359—368.
162. Марчук Г. И. Методы вычислительной математики.— М.: Наука, 1980.— 536 с.
163. Марчук Г. И., Кузнецов Ю. А. К вопросу об оптимальных итерационных процессах // ДАН СССР.— 1968.— Т. 181, № 6.— С. 1331—1334.
164. Маунг Чжо Ньюн, Шарыгин И. Ф. Оптимальные кубатурные формулы на классах $D_2^{1,c}$ и D_2^{1,l_1} // Вопр. вычислительной и прикл. математики.— Ташкент: Ин-т кибернетики АН УзССР, 1971.— С. 22—27.
165. Маюрова И. В., Стронгин Р. Г. Минимизация многоэкстремальных функций, имеющих разрывы // ЖВМиМФ.— 1984.— Т. 24, № 12.— С. 1789—1798.
166. Михайлов Г. А. Дисперсия векторных алгоритмов метода Монте-Карло // ДАН СССР.— 1980.— Т. 253, № 5.— С. 1047—1050.

167. Михайлов Г. А. Оптимизация векторных алгоритмов метода Монте-Карло // ДАН СССР.— 1981.— Т. 260, № 1.— С. 26—31.
168. Михайлов Г. А. Минимаксная теория весовых методов Монте-Карло // ЖВМиМФ.— 1984.— Т. 24, № 9.— С. 1294—1302.
169. Михалевич В. С. Последовательные алгоритмы оптимизации и их применение. I, II // Кибернетика.— 1965.— № 1.— С. 45—56;— № 2.— С. 85—88.
170. Михалевич В. С., Кукса А. И. Методы последовательной оптимизации.— М.: Наука, 1983.— 208 с.
171. Многокритериальные задачи принятия решений/Под ред. Д. М. Гвишиани, С. В. Емельянова.— М.: Машиностроение, 1978.— 191 с.
172. Моисеев Н. Н. Элементы теории оптимальных систем.— М.: Наука, 1975.— 528 с.
173. Моисеев Н. Н. Математические задачи системного анализа.— М.: Наука, 1981.— 488 с.
174. Моисеев Н. Н., Иванилов Ю. П., Столярова Е. М. Методы оптимизации.— М.: Наука, 1978.— 351 с.
175. Молодцов Д. А. Регуляризация множества точек Парето // ЖВМиМФ.— 1978.— Т. 18, № 3.— С. 597—602.
176. Молодцов Д. А. Устойчивость и регуляризация принципов оптимальности // ЖВМиМФ.— 1980.— Т. 20, № 5.— С. 1115—1129.
177. Молчанова Е. С. Построение оптимальных алгоритмов численного интегрирования для одного класса функций.— Деп. ВИНИТИ 03.01.86, № 104—В.— 10 с.
178. Морозов В. А. О принципе оптимальности невязки при приближенном решении уравнений с нелинейными операторами // ЖВМиМФ.— 1974.— Т. 14, № 4.— С. 819—827.
179. Морозов В. А., Гребенников А. И. Об оптимальном приближении операторов // ДАН СССР.— 1975.— Т. 223, № 6.— С. 1307—1310.
180. Морозов В. В., Сухарев А. Г., Фёдоров В. В. Исследование операций в задачах и упражнениях.— М.: Высшая школа, 1986.— 288 с.
181. Моцкус Й. Б. Многоэкстремальные задачи в проектировании.— М.: Наука, 1967.— 215 с.
182. Моцкус Й. Б. О байесовых методах поиска экстремума // Автоматика и вычисл. техн.— 1972.— № 3.— С. 53—62.
183. Налимов В. В., Черпова Н. А. Статистические методы планирования экстремальных экспериментов.— М.: Наука, 1965.— 340 с.
184. Неймарк Ю. И., Стронгин Р. Г. Информационный подход к задаче поиска экстремума функций // Изв. АН СССР. Сер. техн. киберн.— 1966.— № 1.— С. 17—26.
185. Немировский А. С., Нестеров Ю. Е. Оптимальные методы гладкой выпуклой минимизации // ЖВМиМФ.— 1985.— Т. 25, № 3.— С. 356—369.
186. Немировский А. С., Юдин Д. Б. Сложность задач и эффективность методов оптимизации.— М.: Наука, 1979.— 384 с.
187. Немировский А. С., Юдин Д. Б. Информационная сложность математического программирования // Изв. АН СССР. Сер. техн. киберн.— 1983.— № 1.— С. 88—117.

188. Нефёдов В. Н. Об аппроксимации множества Парето // ЖВМиМФ.— 1984.— Т. 24, № 7.— С. 993—1007.
189. Нефёдов В. Н. Отыскание глобального максимума функции нескольких переменных на множестве, заданном ограничениями типа неравенств // ЖВМиМФ.— 1987.— Т. 27, № 1.— С. 35—51.
190. Никольский С. М. К вопросу об оценках приближений квадратурными формулами // УМН.— 1950.— Т. 5, вып. 2(36).— С. 165—177.
191. Никольский С. М. Квадратурные формулы.— М.: Наука, 1979.— 256 с.
192. Ортега Дж., Рейнболдт В. Итерационные методы решения нелинейных систем уравнений со многими неизвестными.— М.: Мир, 1975.— 560 с.
193. Осипенко К. Ю. Оптимальная интерполяция аналитических функций // Мат. заметки.— 1972.— Т. 12, № 4.— С. 465—476.
194. Осипенко К. Ю. Наилучшее приближение функций по информации об их значениях в конечном числе точек // Мат. заметки.— 1976.— Т. 19, № 1.— С. 29—40.
195. Остапенко О. С. Об оптимальных алгоритмах минимизации функций в классах $C^n_{1,L,N}$, $C^n_{1,L,N,\varepsilon}$, $\widetilde{C}^n_{1,L,N,\delta}$ // Кибернетика.— 1983.— № 5.— С. 88—95.
196. Островский А. Решение уравнений и систем уравнений.— М.: ИЛ, 1963.— 219 с.
197. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность.— М.: Мир, 1985.— 512 с.
198. Певный А. Б. Об оптимальных стратегиях поиска максимума функции с ограниченной старшей производной // ЖВМиМФ.— 1982.— Т. 22, № 5.— С. 1061—1066.
199. Пиявский С. А. Алгоритмы отыскания абсолютного минимума функций // Семинар. Теория оптимальных решений, вып. 2.— Киев: Изд-во ИК АН УССР, 1967.— С. 13—24.
200. Пиявский С. А. Об оптимизации сетей // Изв. АН СССР. Сер. техн. киберн.— 1968.— № 1.— С. 68—80.
201. Пиявский С. А. Один алгоритм отыскания абсолютного экстремума функции // ЖВМиМФ.— 1972.— Т. 12, № 4.— С. 888—896.
202. Пласкота Л. Оптимальная линейная информация в задаче восстановления глобального максимума вещественной функции // ЖВМиМФ.— 1986.— Т. 26, № 6.— С. 934—938.
203. Подиновский В. В., Гаврилов В. М. Оптимизация по последовательно применяемым критериям.— М.: Сов. радио, 1975.— 192 с.
204. Подиновский В. В., Ногин В. Д. Парето-оптимальные методы решения многокритериальных задач.— М.: Наука, 1982.— 256 с.
205. Подобедов В. Е. Оптимальный алгоритм поиска экстремума липшицевых функций при произвольной линейной информации // Системное программирование и вопросы оптимизации.— М.: Изд-во Моск. ун-та, 1987.— С. 180—187.
206. Позиционные игры/Под ред. Н. Н. Воробьева, И. Н. Врублевской.— М.: Наука, 1967.— 524 с.
207. Позняк А. С., Цыпкин Я. З. Оптимальные на классе ал-

горитмы оптимизации при наличии коррелированных помех // ЖВМиМФ.— 1984.— Т. 24, № 6.— С. 806—822.
208. Поляк Б. Т. Введение в оптимизацию.— М.: Наука, 1983.— 384 с.
209. Поляк Б. Т., Цыпкин Я. З. Оптимальные псевдоградиентные алгоритмы адаптации // Автоматика и телемехан.— 1980.— № 8.— С. 74—84.
210. Попов Н. М. Аппроксимация множества полуэффективных точек при декомпозиции задач проектирования // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1980.— № 4.— С. 43—48.
211. Попов Н. М. Приближенное решение многокритериальных задач с функциональными ограничениями // ЖВМиМФ.— 1986.— Т. 26, № 10.— С. 1468—1481.
212. Попов Н. М. Численные алгоритмы многокритериальной оптимизации // Системное программирование и вопросы оптимизации.— М.: Изд-во Моск. ун-та, 1987.— С. 155—168.
213. Пшеничный Б. Н. Необходимые условия экстремума.— М.: Наука, 1969.— 152 с.
214. Пшеничный Б. Н. Выпуклый анализ и экстремальные задачи.— М.: Наука, 1980.— 320 с.
215. Пшеничный Б. Н., Данилин Ю. М. Численные методы в экстремальных задачах.— М.: Наука, 1975.— 320 с.
216. Растригин Л. А. Статистические методы поиска.— М.: Наука, 1968.— 376 с.
217. Растригин Л. А. Случайный поиск — проблемы, пути и перспективы // Вопросы кибернетики. Проблемы случайного поиска.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1973.— С. 3—17.
218. Растригин Л. А. Системы экстремального управления.— М.: Наука, 1974.— 632 с.
219. Растригин Л. А. Случайный поиск — специфика, этапы истории и предрассудки // Вопросы кибернетики. Проблемы случайного поиска.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1978.— С. 3—17.
220. Роджерс К. Укладки и покрытия.— М.: Мир, 1968.— 135 с.
221. Рубальский Г. Б. Поиск экстремума унимодальной функции одной переменной на неограниченном множестве // ЖВМиМФ.— 1982.— Т. 22, № 1.— С. 10—16.
222. Рышков С. С. Эффективизация одного метода Давенпорта в теории покрытий // ДАН СССР.— 1967.— Т. 175, № 2.—С. 303—305.
223. Рышков С. С., Барановский Е. П. C-типы n-мерных решеток и пятимерные примитивные параллелоэдры (с приложением к теории покрытий) // Тр. МИАН СССР.— 1976.— Т. 137.— 132 с.
224. Саати Т. Целочисленные методы оптимизации и связанные с ними экстремальные проблемы.— М.: Мир, 1973.— 304 с.
225. Садовников Ю. Ю. Операторный метод глобальной оптимизации // Кибернетика.— 1986.— № 2.— С. 39—43.
226. Самарский А. А. Теория разностных схем.— М.: Наука, 1977.— 656 с.
227. Самарский А. А. Ввведение в численные методы.— М.: Наука, 1982.— 272 с.
228. Смоляк С. А. Об оптимальном восстановлении функций и функционалов от них:— Дис. ... канд. физ.-мат. наук. М.: МГУ, 1965.

229. Соболев С. Л. Введение в теорию кубатурных формул.— М.: Наука, 1974.— 808 с.
230. Соболев С. Л. Коэффициенты оптимальных квадратурных формул // ДАН СССР.— 1977.— Т. 235, № 1.— С. 34—37.
231. Соболь И. М. Многомерные квадратурные формулы и функции Хаара.— М.: Наука, 1969.— 288 с.
232. Соболь И. М. Численные методы Монте-Карло.— М.: Наука, 1973.— 312 с.
233. Соболь И. М. Об оценке точности простейшего многомерного поиска // ДАН СССР.— 1982.— Т. 266, № 3.— С. 569—572.
234. Соболь И. М., Статников Р. Б. Выбор оптимальных параметров в задачах со многими критериями.— М.: Наука, 1981.— 110 с.
235. Современное состояние теории исследования операций/Под ред. Н. Н. Моисеева.— М.: Наука, 1979.— 464 с.
236. Сорокина Т. Г., Сорокин И. М. Минимизация многоэкстремальных функций // Математические методы оптимизации и управления в сложных системах.— Калинин: Изд-во КГУ, 1981.— С. 60—68.
237. Стригуль О. И. Оптимальные алгоритмы поиска минимакса для функций некоторых классов // Кибернетика.— 1983.— № 5.— С. 126.
238. Стронгин Р. Г. Численные методы в многоэкстремальных задачах.— М.: Наука, 1978.— 240 с.
239. Стронгин Р. Г., Маркин Д. Л. Минимизация многоэкстремальных функций при невыпуклых ограничениях // Кибернетика.— 1986.— № 4.— С. 64—69.
240. Сухарев А. Г. Об оптимальных стратегиях поиска экстремума // ЖВМиМФ.— 1971.— Т. 11, № 4.— С. 910—924.
241. Сухарев А. Г. Наилучшие стратегии последовательного поиска экстремума // ЖВМиМФ.— 1972.— Т. 12, № 1.— С. 35—50.
242. Сухарев А. Г. Оптимальный поиск экстремума.— М.: Изд-во Моск. ун-та, 1975.— 100 с.
243. Сухарев А. Г. Оптимальный поиск корня функции, удовлетворяющей условий Липшица // ЖВМиМФ.— 1976.— Т. 16, № 1.— С. 20—29.
244. Сухарев А. Г. Оптимальные пассивные и последовательно-оптимальные алгоритмы построения наилучших аппроксимаций для функций, удовлетворяющих условию Липшица // ДАН СССР.— 1976.— Т. 231, № 4.— С. 814—817.
245. Сухарев А. Г. Решение многошаговой антагонистической игры аппроксимации // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1977.— № 2.— С. 43—57.
246. Сухарев А. Г. Оптимальный метод построения наилучших равномерных приближений для функций некоторого класса // ЖВМиМФ.— 1978.— Т. 18, № 2.— С. 302—313.
247. Сухарев А. Г. Оптимальные формулы численного интегрирования для некоторых классов функций многих переменных // ДАН СССР.— 1979.— Т. 246, № 2.— С. 282—285.
248. Сухарев А. Г. Глобальный экстремум и методы его отыскания // Математические методы в исследовании операций.— М.: Изд-во Моск. ун-та, 1981.— С. 4—37.
249. Сухарев А. Г. Оптимальные и последовательно-оптимальные алгоритмы в задачах численного анализа // Математиче-

ские методы в исследовании операций.— М.: Изд-во Моск. ун-та, 1981.— С. 54—68.
250. Сухарев А. Г. Построение оптимального на один шаг стохастического алгоритма поиска экстремума // ЖВМиМФ.— 1981.— Т. 21, № 6.— С. 1385—1401.
251. Сухарев А. Г. Об одном подходе к исследованию эффективности квадратурных формул // Кибернетика.— 1981.— № 6.— С. 75—80.
252. Сухарев А. Г. К вопросу о построении оптимальных квадратур для функций многих переменных // Кибернетика.— 1982.— № 1.— С. 7—11.
253. Сухарев А. Г. Об оптимальных методах решения многокритериальных задач // Изв. АН СССР. Сер. техн. киберн.— 1982.— № 3.— С. 67—73.
254. Сухарев А. Г. К вопросу о совпадении погрешностей в классах пассивных и последовательных алгоритмов // ЖВМиМФ.— 1985.— Т. 25, № 2.— С. 295—298.
255. Сухарев А. Г., Тимохов А. В., Фёдоров В. В. Курс методов оптимизации.— М.: Наука, 1986.— 328 с.
256. Сухарев А. Г., Фёдоров В. В. Минимаксные задачи и минимаксные алгоритмы.— М.: Изд-во Моск. ун-та, 1979.— 50 с.
257. Сухарев А. Г., Фёдоров В. В. Об оптимальном отыскании максимума функции минимума при связанных переменных // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1981.— № 1.— С. 45—50.
258. Сухарев А. Г., Фёдоров В. В. Оптимальные алгоритмы глобального поиска в минимаксных задачах // Вопросы кибернетики. Модели и методы глобальной оптимизации.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1985.— С. 70—79.
259. Сухарев А. Г., Чуян О. Р. О применении достаточных условий совпадения погрешностей в классах пассивных и последовательных алгоритмов.— Деп. ВИНИТИ 31.05.84, № 3535—84.— 10 с.
260. Сысоев В. В., Петров В. А. Разработка программы оптимизации многоэкстремальных функций методом ψ-преобразования // Эконом. и матем. методы.— 1976.— Т. 12, № 1.— С. 178—184.
261. Танана В. П. Методы решения операторных уравнений.— М.: Наука, 1981.— 156 с.
262. Тарасова В. П. Оптимальные стратегии поиска области наибольших значений для некоторого класса функций // ЖВМиМФ.— 1978.— Т. 18, № 4.— С. 886—896.
263. Тарасова В. П. Оптимальные алгоритмы поиска отрезка наибольших значений для некоторого класса функций // ЖВМиМФ.— 1981.— Т. 21, № 5.— С. 1108—1115.
264. Тарасова В. П. Оптимальный поиск экстремума локально-унимодальных функций // Кибернетика.— 1984.— № 1.— С. 65—68.
265. Тимонов Л. Н. Алгоритм поиска глобального экстремума // Изв. АН СССР. Сер. техн. киберн.— 1977.— № 3.— С. 53—60.
266. Тихомиров В. М. Наилучшие методы приближения и интерполирования дифференцируемых функций в пространстве $C[-1, 1]$ // Мат. сб.— 1969.— Т. 80 (122), № 2(10).— С. 290—304.

267. Тихомиров В. М. Некоторые вопросы теории приближений.— М.: Изд-во Моск. ун-та, 1976.— 304 с.
268. Тихонов А. Н. О решении некорректно поставленных задач и методе регуляризации // ДАН СССР.— 1963.— Т. 151, № 3.— С. 501—504.
269. Тихонов А. Н. О регуляризации некорректно поставленных задач // ДАН СССР.— 1963.— Т. 153, № 1.— С. 49—52.
270. Тихонов А. Н. О нелинейных уравнениях первого рода // ДАН СССР.— 1965.— Т. 161, № 5.— С. 1023—1026.
271. Тихонов А. Н., Арсенин В. Я. Методы решения некорректных задач.— М.: Наука, 1979.— 285 с.
272. Тихонов А. Н., Гайсарян С. С. О выборе оптимальных сеток при приближенном вычислении квадратур // ЖВМиМФ.— 1969.— Т. 9, № 5.— С. 1170—1176.
273. Торохтий А. П. Об оптимальных на классах сеток методах вычисления квадратур // ЖВМиМФ.— 1983.— Т. 23, № 1.— С. 29—38.
274. Трауб Дж. Итерационные методы решения уравнений.— М.: Мир, 1985.— 264 с.
275. Трауб Дж., Васильковский Г., Вожьняковский Х. Информация, неопределенность, сложность.— М.: Мир, 1988.— 184 с.
276. Трауб Дж., Вожьняковский Х. Общая теория оптимальных алгоритмов.— М.: Мир, 1983.— 384 с.
277. Уайлд Д. Методы поиска экстремума.— М.: Наука, 1967.— 268 с.
278. Устюжанинов В. Г. О случайном поиске оптимальной системы признаков // Вычислительные системы, вып. 77. Автоматизация проектирования в микроэлектронике. Теория, методы, алгоритмы.— Новосибирск: Наука, 1978.— С. 42—52.
279. Устюжанинов В. Г. Предельные возможности алгоритмов случайного поиска // Автоматика и вычисл. техн.— 1980.— № 1.— С. 71—77.
280. Устюжанинов В. Г. Об эффективности случайного поиска // Автоматика и вычисл. техн.— 1980.— № 3.— С. 70—74.
281. Устюжанинов В. Г. Поиск максимума функции с ограничением Липшица // Проблемы случайного поиска.— Рига: Зинатне, 1980, вып. 8.— С. 135—148.
282. Устюжанинов В. Г. О существовании и эффективности оптимальных алгоритмов случайного поиска // Проблемы случайного поиска.— Рига: Зинатне, 1981, вып. 9.— С. 9—29.
283. Устюжанинов В. Г. Возможности случайного поиска в решении дискретных экстремальных задач // Кибернетика.— 1983.— № 2.— С. 64—77.
284. Устюжанинов В. Г. Случайный поиск в непрерывных задачах глобальной оптимизации // Вопросы кибернетики. Модели и методы глобальной оптимизации.— М.: Научн. совет АН СССР по компл. пробл. «Кибернетика», 1985.— С. 37—45.
285. Фёдоров В. В. Теория оптимального эксперимента.— М.: Наука, 1971.— 312 с.
286. Фёдоров В. В. Численные методы максимина.— М.: Наука, 1979.— 280 с.
287. Финни Д. Введение в теорию планирования экспериментов.— М.: Наука, 1970.— 288 с.

288. Химмельблау Д. Прикладное нелинейное программирование.— М.: Мир, 1975.— 534 с.
289. Цермело Э. Об одном применении теории множеств к теории шахматной игры // Матричные игры.— М.: Физматгиз, 1961.— С. 167—172.
290. Черноусько Ф. Л. Оптимальный алгоритм поиска корня функции, вычисляемой приближенно // ЖВМиМФ.— 1968.— Т. 8, № 4.— С. 705—724.
291. Черноусько Ф. Л. Об оптимальном поиске экстремума унимодальных функций // ЖВМиМФ.— 1970.— Т. 10, № 4.— С. 922—933.
292. Черноусько Ф. Л. Об оптимальном поиске минимума выпуклых функций // ЖВМиМФ.— 1970.— Т. 10, № 6.— С. 1355—1366.
293. Черноусько Ф. Л., Меликян А. А. Игровые задачи управления и поиска.— М.: Наука, 1978.— 270 с.
294. Чичинадзе В. К. Решение невыпуклых нелинейных задач оптимизации.— М.: Наука, 1983.— 256 с.
295. Чуян О. Р. Об оптимальных алгоритмах поиска экстремума дифференцируемых функций // Вестн. Моск. ун-та. Сер. вычисл. матем. и киберн.— 1984.— № 3.— С. 28—34.
296. Чуян О. Р. Оптимальный одношаговый алгоритм максимизации дважды дифференцируемых функций // ЖВМиМФ.— 1986.— Т. 26, № 3.— С. 383—397.
297. Шалтянис В. Р. Об одном методе многоэкстремальной оптимизации // Автоматика и вычисл. техн.— 1971.— № 3.— С. 33—38.
298. Шокин Ю. И. Интервальный анализ.— Новосибирск: Наука, 1981.— 112 с.
299. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения.— Киев: Наук. думка, 1979.— 119 с.
300. Юдин Д. Б. Вычислительные методы многокритериальной оптимизации // Изв. АН СССР. Сер. техн. киберн.— 1983.— № 4.— С. 3—16.
301. Aird T. J., Rice J. R. Systematic search in high dimensional sets // SIAM J. Numer. Anal.— 1977.— V. 14, № 2.— P. 296—312.
302. Anderssen R. S. Global optimization // Optimization.— St. Lucia, Australia: Univ. of Queensland press, 1972.— P. 26—48.
303. Anderssen R. S., Bloomfield P. Properties of the random search in global optimization // J. Optimization Theory and Appl.— 1975.— V. 16, № 5/6.— P. 383—398.
304. Archetti F., Betro B. Some remarks on dimensionality reduction techniques in global optimization problems.— Università di Pisa, Pisa, Italy, Dipartimento di Ricerca Operativa e Scienze Statistiche, Report A—49, 1978.
305. Archetti F., Betro B. On the effectiveness of uniform random sampling in global optimization problems.— Università di Pisa, Pisa, Italy, Dipartimento di Ricerca Operativa e Scienze Statistiche, Report A—51, 1978.
306. Archetti F., Betro B. A stopping criterion for global optimization algorithms.— Università di Pisa, Pisa, Italy, Dipartimento di Ricerca Operativa e Scienze Statistiche, Report A—56, 1978.

307. Archetti F., Betro B. A probabilistic algorithm for global optimization // Calcolo.— 1979.— V. 16, № 3.— P. 335—343.
308. Archetti F., Betro B. Stochastic models and optimization // Bolletino della Unione Matematica Italiana.— 1980.— V. 5, № 17—A.— P. 225—301.
309. Archetti F., Szegö G. P. Global optimization algorithms // Nonlinear optimization, theory and algorithms.— Boston: Birkhauser, 1980.— P. 429—469.
310. Avriel M., Wilde D. J. Optimality proof for the symmetric Fibonacci search technique // Fibonacci Quart.— 1966.— V. 4, № 3.— P. 265—269.
311. Avriel M., Wilde D. J. Golden block search for the maximum of unimodal functions // Manage. Sci.— 1968.— V. 14, № 5.— P. 307—319.
312. Baba N. Convergence of a random optimization method for constrained optimization problems // J. Optimization Theory and Appl.— 1981.— V. 33, № 4.— P. 451—461.
313. Baba N. A hybrid algorithm for finding a global minimum // Int. J. Control.— 1983.— V. 37, № 5.— P. 929—942.
314. Babenko V. F. On the optimal error bound for cubature formulae on certain classes of continuous functions // Analysis Mathematica.— 1977.— V. 3, № 1.— P. 3—9.
315. Basso P. Iterative methods for the localization of the global maximum // SIAM J. Numer. Anal.— 1982.— V. 19, № 4.— P. 781—792.
316. Beamer J. H., Wilde D. J. Time delay in minimax optimization of unimodal functions of one variable // Manage Sci.— 1969.— V. 15, № 9.— P. 528—538.
317. Beamer J. H., Wilde D. J. Minimax optimization of unimodal functions by variable block search // Manage.Sci.— 1970.— V. 16, № 9.— P. 529—541.
318. Beamer J. H., Wilde D. J. Minimax optimization of a unimodal function by variable block derivative search with time delay // J. Comb. Th.— 1971.—V. 10, № 2.— P. 160—173.
319. Betro B. Bayesian testing of nonparametric hypotheses and its application to global optimization // J. Optimization Theory and Appl.— 1984.— V. 42, № 1.— P. 31—50.
320. Boender C. G. E., Rinnooy Kan A. H. G., Timmer G. T., Stougie L. A stochastic method for global optimization // Math. Programming.— 1982.— V. 22, № 2.— P. 125—140.
321. Bojanov B. D. Comparision theorems in optimal recovery // Optimal Algorithms.— Sofia: BAS, 1986.— P. 15—50.
322. Bojanov B. D., Chernogorov V. G. An optimal interpolation formula // J. Approx. Theory.— 1977.— V. 20, № 3.— P. 264—274.
323. Booth R. S. Location of zeros of derivatives // SIAM J. Appl. Math.— 1967.— V. 15, № 6.— P. 1496—1501.
324. Booth R. S. Location of zeros of derivatives, II // SIAM J. Appl. Math.— 1969.— V. 17, № 2.— P. 409—415.
325. Boult T. Some examples and applications of information-based complexity // Optimal Algorithms.— Sofia: BAS, 1986.— P. 51—64.

326. Brooks S. H. A discussion of random methods for seeking maxima // Operations Res.— 1952.— V. 6, № 2.— P. 244—251.
327. Brown K. M. Solution of simultaneous nonlinear equations // Comm. Assoc. Comput. Mach.— 1967.— V. 10, № 11.— P. 728—729.
328. Brown K. M. A quadratically convergent Newton-like method based upon Gaussian elimination // SIAM J. Numer. Analysis.— 1969.— V. 6, № 4.— P. 560—569.
329. Clarke F. H. Optimization and nonsmooth analysis.— New York: Wiley, 1983.— 308 p.
330. Converse A. O. The use of uncertainty in a simultaneous search // Operations Res.— 1967.— V. 15, № 6.— P. 1088—1095.
331. Coxeter H. The classification of zonohedra by means of projective diagrams // Journal de mathématiques pures et appliquées.— 1962.— V. 41, № 2.— P. 137—156.
332. De Boor C. On writing an automatic integration algorithm // Matematical software.— New York, London: Academic Press, 1971.— P. 201—209.
333. De Boor C. CARDE: An algorithm for numerical quadrature // Mathematical software.— New York, London: Academic Press, 1971.— P. 417—449.
334. De Boor C. Computational aspects of optimal recovery // Optimal estimation in approximation theory.— New York, London: Plenum Press, 1977.— P. 69—91.
335. De Boor C., Rice J. R. An adaptive algorithm for multivariate approximation // J. Approx. Theory.— 1979.— V. 25, № 4.— P. 337—360.
336. Devroye L. Progressive global random search of continuous functions // Math. Programming.— 1978.— V. 15, № 3.— P. 330—342.
337. Dixon L. C. W. Reflections on nondifferentiable optimization. Part 1. Ball gradient // J. Optimization Theory and Appl.— 1980.— V. 32, № 2.— P. 123—133.
338. Dixon L. C. W., Gaviano M. Reflections on nondifferentiable optimization. Part 2. Convergence // J. Optimization Theory and Appl.— 1980.— V. 32, № 3.— P. 259—276.
339. Dixon V. A. Numerical quadrature: A survey of the available algorithms // Software for numerical mathematics.— London, New York: Academic Press, 1974.— P. 105—137.
340. Eichhorn B. H. On sequential search // Selected statistical papers. V. 1.— Amsterdam: Math. Centrum, 1968.— P. 81—95.
341. Einarsson B. Testing and evaluation of some subroutines for numerical quadrature // Software for numerical mathematics.— London, New York: Academic Press, 1974.— P. 149—157.
342. Fine T. Optimum search for the location of the maximum of a unimodal function // IEEE Trans. on Information Theory.— 1966.— V. 12, № 2.— P. 103—111.
343. Fisher S. D., Micchelli C. A. Optimal sampling of holomorphic functions // Amer. J. Math.— 1984.— V. 106, № 3.— P. 593—609.
344. Gaffney P. W. The range of possible values of $f(x)$ // Computer Science and Systems Division Rep., AERE, Harwell, Oxfordshire, 1977.

345. Gaffney P. W. To compute the optimal interpolation formula // Math. Comp.— 1978.— V. 32, № 143.— P. 763—777.
346. Gaffney P. W., Powell M. J. D. Optimal interpolation // Numerical analysis, Lecture Notes in Math., v. 506.— Berlin, New York: Springer Verlag, 1976.— P. 90—100.
347. Gal S. Sequential minimax search for a maximum when prior information is available // SIAM J. Appl. Math.— 1971.— V. 21, № 4.— P. 590—595.
348. Gal S. Multidimensional minimax search for a maximum // SIAM J. Appl. Math.— 1972.— V. 23, № 4.— P. 513—526.
349. Gal S. Optimal sequential and parallel search for finding a root // J. Comb. Th. Ser. A.— 1977.— V. 23, № 1.— P. 1—14.
350. Gal S., Micchelli C. A. Optimal sequential and nonsequential procedures for evaluating a functional // Applicable Analysis.— 1980.— V. 10, № 2.— P. 105—120.
351. Gaviano M. Some general results on the convergence of random search algorithms in minimisation problems // Towards global optimization.— Amsterdam: North-Holland, 1975.— P. 149—157.
352. Goldstein A. A. Optimization of Lipschitz continuous functions // Math. Programming.— 1977.— V. 13, № 1.— P. 14—22.
353. Golomb M. Interpolation operators as optimal recovery schemes for classes of analytic functions // Optimal estimation in approximation theory.— New York, London: Plenum Press, 1977.— P. 93—138.
354. Gross O. A class of discrete type minimization problems // Rand Corporation. Research Memorandum № 1644. —1956.
355. Gross O., Johnson S. Sequential minimax search for a zero of a convex function // Math. Tables and Other Aids to Computation.— 1959.— V. 13, № 65.— P. 44—51.
356. Haber S. Adaptive integration and improper integrals // Math. Comp.— 1975.— V. 29, № 131.— P. 806—809.
357. Hansen E. R. Global optimization using interval analysis: The one-dimensional case // J. Optimization Theory and Appl.— 1979.— V. 29, № 3.— P. 331—344.
358. Hansen E. R. Global optimization using interval analysis: The multi-dimensional case // Numer. Math.— 1980.— V. 34, № 3.— P. 247—270.
359. Hansen E. R. Global optimization with data perturbations // Comput. and Oper. Res.— 1984.— V. 11, № 2.— P. 97—104.
360. Heindl G. Optimal quadrature of convex functions // Numerical integration. International series of numerical mathematics, v. 57.— Basel: Birkhauser Verlag, 1982. P. 138—147.
361. Heyman M. Optimal simultaneous search for the maximum by the principle of statistical information // Operations Res.— 1968.— V. 16, № 6.— P. 1194—1205.
362. Hofman K. L. A method for globally minimizing concave functions over convex sets // Math. Programming.— 1981.— V. 20, № 1.— P. 22—32.
363. Hyafil L. Optimal search for the zero of the $(n-1)$st derivative.— IRIA/LABORIA Rep. № 247, 1977.

364. Johnson S. J. Best exploration for maximum is Fibonaccian.— Rand Corporation Research Memorandum, RM— 1590, Santa-Monica, 1955.
365. Kacewicz B. Z. On the optimal error of algorithms for solving a scalar autonomus ODE // BIT.— 1982.— V. 22, № 4.— P. 503—518.
366. Kacewicz B. Z. Optimality of Euler-integral information for solving a scalar autonomus ODE // BIT.— 1983.— V. 23, № 2.— P. 217—230.
367. Kacewicz B. Z. How to increase the order to get minimal-error algorithms for systems of ODE // Num. Math.—1984.— V. 45, № 1.— P. 93—104.
368. Kahaner D. K. Comparison of numerical quadrature formulas // Mathematical software.— New York, London: Academic Press, 1971.— P. 229—259.
369. Karp R. M., Miranker W. L. Parallel minimax search for a maximum // J. Comb. Th.— 1968.— V. 4, № 1.— P. 19—35.
370. Kaupe A. F. On optimal search techniques // Comm. Assoc. Comput. Mach.— 1964.— V. 7, № 1.— P. 38.
371. Kiefer J. Sequential minimax search for a maximum // Proc. Amer. Math. Soc.— 1953.— V. 4, № 3.— P. 502—506.
372. Kiefer J. Optimum sequential search and approximation methods under minimum regularity assumptions // J. Soc. Indust. Appl. Math.— 1957.— V. 5, № 3.— P. 105—136.
373. Krolak P. D. A property of the Krolak-Cooper extension of Fibonaccian search // SIAM Review.— 1966.— V. 8, № 4.— P. 510—517.
374. Krolak P. D. Further extensions of Fibonaccian search to nonlinear programming problems // SIAM J. Control.— 1968.— V. 6, № 2.— P. 258—265.
375. Krolak P. D., Cooper L. An extension of Fibonaccian search to several variables // Comm. Assoc. Comput. Mach.— 1963.— V. 6, № 10.— P. 639—641.
376. Kushner H. A versatile stochastic model of a function of unknown and time varying form // J. Math. Anal. and Appl.— 1962.— V. 5, № 1.— P. 150—167.
377. Kushner H. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise // Trans. ASME. Ser. D. J. Basic Eng.— 1964.— V. 86, № 1.— P. 97—105.
378. Lootsma F. Comparative performance evaluation, experimental design, and generation of test problems in nonlinear optimization // Computational mathematical programming.— Berlin: Springer-Verlag, 1985.— P. 249—260.
379. Lyness J. N. Notes on adaptive Simpson quadrature // J. Assoc. Comput. Mach.— 1969.— V. 16, № 3.— P. 483—495.
380. Lyness J. N., Kaganove J. J. Comments on the nature of automatic quadrature routines // ACM Trans. Math. Software.— 1976.— V. 2, № 1.— P. 65—81.
381. Madsen K. Minimax solution of non-linear equations without calculating derivatives // Nondifferentiable optimization, Mathematical Programming Study, 3.— Amsterdam: North-Holland, 1975.— P. 110—126.

382. Malcolm M. A., Simpson R. B. Local versus global strategies for adaptive quadrature // ACM Trans. Math. Software.— 1975.— V. 1, № 2.— P. 129—146.
383. Mancini L., Mc Cormick G. P. Bounding global minima // Math. Oper. Res.— 1976.— V. 1, № 1.— P. 50—53.
384. Mancini L., Mc Cormick G. P. Bounding global minima with interval arithmetic // Operations Research.— 1979.— V. 26, № 4.— P. 743—754.
385. Mc Cormick G. P. Attempts to calculate global solutions of problems that may have local minima // Numerical methods for non-linear optimization.— London, New York: Academic Press, 1972.— P. 209—221.
386. Mc Cormick G. P. Locating an isolated global minimizer of a constrained nonconvex programm // Math. Oper. Res.— 1980.— V. 5, № 3.— P. 435—443.
387. Mc Cormick G. P. Nonlinear programming. Theory, algorithms and applications.— New York: Wiley, 1983.— 461 p.
388. Melkman A. A., Micchelli C. A. Optimal estimation of linear operators in Hilbert spaces from inaccurate data // SIAM J. Numer. Anal.— 1979.— V. 16, № 1.— P. 87—105.
389. Micchelli C. A. Optimal estimation of smooth functions from inaccurate data.— IBM Research Report 7024,Yorktown Heights, New York, 1978.
390. Micchelli C. A. Orthogonal projections are optimal algorithms // J. Approx. Theory.— 1984.— V. 40, № 2.— P. 101—110.
391. Micchelli C. A., Miranker W. L. High order search methods for finding roots // J. Assoc. Comput. Mach.— 1975.— V. 22, № 1.— P. 51—60.
392. Micchelli C. A., Rivlin T. J. A survey of optimal recovery // Optimal estimation in approximation theory.— New York, London: Plenum Press, 1977.— P. 1—54.
393. Micchelli C. A., Rivlin T. J., Winograd S. The optimal recovery of smooth functions // Numer. Math.— 1976.— V. 260, № 2.— P. 191—200.
394. Milanese M., Tempo R. Optimal algorithms theory for robust estimation and prediction // IEEE Trans. Automat. Contr.— 1985.— V. AC—30, № 8.— P. 730—738.
395. Milne-Thomson L. M. The calculas of finite differences.— London: Mac Millan, 1933.— 528 p.
396. Miranker W. L. Parallel methods of approximating the root of a function // IBM J. of Research and Development.— 1969.— V. 13, № 3.— P. 297—301.
397. Miranker W. L. Parallel methods for solving equations.— IBM T. J. Watson Research Center Rep. 6545, 1977.
398. Mjelde K. M. Methods of the allocation of limited resources.— New York: Wiley, 1983.— 96 p.
399. Mockus J. On Bayesian methods of seeking the extremum and their applications // Information processing 77.— Amsterdam: North-Holland, 1977.— P. 195—200.
400. Mockus J. The simple Bayesian algorithm for the multidimensional global optimization // Numerical techniques for stochastic systems.— Amsterdam: North-Holland, 1980.— P. 369—377.

401. Mustata C. Best approximation and unique extension of Lipschitz functions // J. Approx. Theory.— 1977.— V. 19, № 3.— P. 222—230.
402. Neuman P. An asymptotically optimal procedure for searching a zero or an extremum of a function if a prior distribution of its location is known // Proc. of a second Prague symposium on asympt. statistics.— Prague: Academia, 1981. P. 291—302.
403. Newman D. J. Location of the maximum on unimodal surfaces // J. Assoc. Comput. Mach.— 1965.— V. 12, № 3.— P. 395—398.
404. Niederreiter H., Mc Curley K. Optimization of functions by quasy-random search methods // Computing.— 1979.— V. 22, № 2.— P. 119—123.
405. Niederreiter H., Peart P. Localization of search in quasi Monte-Carlo methods for global optimization // SIAM J. Sci. Stat. Comput.— 1986.— V. 7, № 2.— P. 660—664.
406. Oliver L. T., Wilde D. J. Symmetric sequential minimax search for a maximum // Fibonacci Quart.— 1964.— V. 2, № 1.— P. 169—175.
407. Performance evaluation of numerical software/Ed. L. G. Fosdick.— Amsterdam: North-Holland, 1979.— 336 p.
408. Pourciau B. H. Analysis and optimization of Lipschitz continuous mappings // J. Optimization Theory and Appl.— 1977.— V. 22, № 3.— P. 311—351.
409. Renegar J. On the complexity of a piecewise linear algorithms for approximating roots of complex polynomials // Math. Programming.— 1985.— V. 32, № 3.— P. 301—318.
410. Renegar J. On the cost of approximating all roots of a complex polynomial // Math. Programming.— 1985.— V. 32, № 3.— P. 319—336.
411. Renegar J. On the efficiency of Newton's method in approximating all zeros of a system of complex polynomials.— Berkeley Mathematical Sciences Research Institute Report, MSRI 00318—86, 1985.
412. Renegar J. Rudiments of an average case complexity theory for piecewise-linear path following algorithms.— Berkeley Mathematical Sciences Research Institute Report, MSRI 03418—86, 1986.
413. Rice J. R. Parallel algorithms for adaptive quadrature — convergence // Proc. IFIP Congress 74, Stokholm.— Amsterdam: North-Holland, 1974.— P. 600—604.
414. Rice J. R. A metalgorithm for adaptive quadrature // J. Assoc. Comput. Mach.— 1975.— V. 22, № 1.— P. 61—82.
415. Rice J. R. Parallel algorithms for adaptive quadrature II — metalgorithm correctness // Acta Informatica.— 1976.— V. 5, № 4.— P. 273—275.
416. Rice J. R. Parallel algorithms for adaptive quadrature III— programm correctness // ACM Trans. Math. Software.— 1976.— V. 2, № 1.— P. 1—30.
417. Rice J. R. Numerical methods, software and analysis.— New York: Mc Graw-Hill, 1983.— 671 p.
418. Rinnooy Kan A. H. G., Boender C. G. E., Timmer G. T. A stochastic approach to global optimization // Computational mathematical programming.— Berlin: Springer-Verlag, 1985.— P. 281—305.

419. Rogers C. A. A note on coverings // Mathematika. A Journal of Pure and Applied Mathematics.— 1957.— V. 4. Part. 1. № 7.— P. 1—6.
420. Sard A. Best approximate integration formulas: best approximation formulas // Amer. J. Math.— 1949.— V. 71, № 1.— P. 80—91.
421. Shapiro H. D. Increasing robustness in global adaptive quadrature through interval selection heuristics // ACM Trans. Math. Software.— 1984.— V. 10, № 2.— P. 117—139.
422. Shapiro R. D., Wilde D. J. Optimal minimax search with unequal block sizes.— Stanford University, Dept. of Operations Res., Technical Report 74—16, 1974.
423. Shapiro R. D., Wilde D. J. Variable block search strategies with prior information.— Stanford University, Dept. of Operations Res., Technical Report 74—17, 1974.
424. Shih W. A new application of incremental analysis in resource allocations // Operational Research Quartely.— 1974.— V. 25, № 4.— P. 587—597.
425. Shub M., Smale S. Computational complexity: on the geometry of polynomials and a theory of cost. Part I // Ann. scient. Éc. Norm. Sup. 4 ser.— 1985.— T. 18, № 1.— P. 107—142; Part II // SIAM J. Comput.— 1986.— V. 15, № 1.— P. 145—161.
426. Shub M., Smale S. On the existence of generally convergent algorithms // J. Complexity.— 1986.— V. 2, № 1.— P. 2—11.
427. Shubert B. O. A sequential method seeking the global maximum of a function // SIAM J. Numer. Anal.— 1972.— V. 9, № 3.— P. 379—388.
428. Shubert B. O. Sequential optimization of multimodal discrete function with bounded rate of change // Manage. Sci.— 1972.— V. 18, № 11.— P. 687—693.
429. Sikorski K. Bisection is optimal // Numer. Math.— 1982.— V. 40, № 1.— P. 111—117.
430. Sikorski K. Optimal solution of nonlinear equations satisfying a Lipschitz condition // Num. Math.— 1984.— V. 43, № 2.— P. 225—240.
431. Smale S. The fundamental theorem of algebra and complexity theory // Bull. Amer. Math. Soc. (N. S.)— 1981.— V. 4, № 1.— P. 1—36.
432. Smale S. On the average number of steps in the simplex method of linear programming // Math. Programming.— 1983.— V. 27, № 3.— P. 241—262.
433. Smale S. The problem of the average speed of the simplex method // Mathematical programming. The state of the art.— Berlin, New York: Springer-Verlag, 1983.— P. 530—539.
434. Smale S. On the efficiency of algorithms of analysis // Bull. Amer. Math. Soc. (N. S.)— 1985.— V. 13, № 2.— P. 87—121.
435. Sobol I. M. On the systematic search in a hypercube // SIAM J. Numer. Anal.— 1979.— V. 16, № 5.— P. 790—793.
436. Sonnevend G. On optimization of algorithms for function minimization // ЖВМ и МФ.— 1977.— Т. 17, № 3.— С. 591—609.

437. S o n n e v e n d G. On the optimization of adaptive algorithms // Operations Research Verfahren, v. 31.— Athenäum, 1978.— P. 581—595.
438. S o n n e v e n d G. Optimal passive and sequential algorithms for the approximation of convex functions in $L_p([0, 1]^s)$, $p = 1, \infty$ // Конструктивная теория функций 81.— София, 1983.— С. 535—542.
439. S o n n e v e n d G. An optimal sequential algorithms for the uniform approximation of convex functions on $[0, 1]^2$ // Applied Mathematics and Optimization.— 1983.— V. 10, № 2.— P. 127—142.
440. S u g i e N. An extension of Fibonaccian searching to multidimensional cases // IEEE Trans. on Automatic Control.— 1964.— V. 9, № 1.— P. 105.
441. S u k h a r e v A. G. A sequentially optimal algorithm for numerical integration // J. Optimization Theory and Appl.— 1979.— V. 28, № 3.— P. 363—373.
442. S u k h a r e v A. G. Optimal algorithms for iterated numerical integration // J. Optimization Theory and Appl.— 1979.— V. 28, № 3.— P. 375—390.
443. S u k h a r e v A. G. On optimal stochastic and deterministic search for global extremum // Internationale Tagung Mathematische Optimierung — Theorie und Anwendungen.— DDR, Ilmenau: Technische Hochschule, 1981.— P. 163—166.
444. S u k h a r e v A. G. On optimal methods in numerical analysis // Computational mathematics, Banach Center Publications, v. 13.— Warsaw: PWN — Polish Scientific Publishers, 1984.— P. 575—587.
445. S u k h a r e v A. G. On the existence of optimal affine methods for approximating linear functionals // J. Complexity.— 1986.— V. 2, № 4.— P. 317—322.
446. T o d d M. J. Optimal dissection of simplices.— Cornell University, Dept. of Operations Res. Report, 1976.
447. Towards global optimization, I/Eds. L. C. W. Dixon, G. P. Szegö.— Amsterdam: North-Holland, 1975.— 472 p.
448. Towards global optimization, II /Eds. L. C. W. Dixon, G. P. Szegö.— Amsterdam: North-Holland, 1978.— 363 p.
449. T r a u b J. F. Complexity of approximately solved problems // J. Complexity.— 1985.— V. 1, № 1.— P. 3—10.
450. T r a u b J. F., W a s i l k o w s k i G. W., W o ź n i a k o w s k i H. Average case optimality for linear problems // Theoret. Comp. Sci.— 1984.— V. 29, № 1.— P. 1—25.
451. T r a u b J. F., W o ź n i a k o w s k i H. Information and computation // Advances in computers, v. 23.— New York, London: Academic Press, 1984.— P. 35—92.
452. T r a u b J. F., W o ź n i a k o w s k i H. On the optimal solution of large linear systems // J. Assoc. Comput. Mach.— 1984.— V. 31, № 3.— P. 545—559.
453. V e i n o t t A. F. Production planning with convex costs: a parametric study // Manage. Sci.— 1966.— V. 12, № 11.— P. 745—777.
454. W a s i l k o w s k i G. W. Some nonlinear problems are as easy as the approximation problem // Comp. and Maths. with Appls.— 1984.— V. 10, № 4/5.— P. 351—363.

455. Wasilkowski G. W. Average case optimality // J. Complexity.— 1985.— V. 1, № 1.— P. 107—117.
456. Wasilkowski G. W., Woźniakowski H. Can adaptation help on the average? // Num. Math.— 1984.— V. 44, № 2.— P. 169—190.
457. Werschulz A. G. Optimal order for approximation of derivatives // J. Comp. Syst. Sci.— 1979.— V. 18, № 3.— P. 213—217.
458. Werschulz A. G. Computational complexity of one-step methods for systems of differential equations // Math. Comp.— 1980.— V. 34, № 149.— P. 155—174.
459. Werschulz A. G. On maximal order for local and global problems // J. Comp. Syst. Sci.— 1981.— V. 23, № 1.— P. 38—48.
460. Werschulz A. G. Does increased regularity lower complexity? // Math. Comp.— 1984.— V. 42, № 165.— P. 69—93.
461. Werschulz A. G. Optimal algorithms for a problem of optimal control // Optimal algorithms.— Sofia: BAS, 1986.— P. 228—234.
462. Wilde D. J. A multivariable dichotomous optimum-seeking method // IEEE Trans. on Automatic Control.— 1965.— V. 10, № 1.— P. 85—87.
463. Wilde D. J., Beightler C. S. Foundations of optimization.— Englewood Cliffs, New Jersey: Prentice-Hall, 1967.— 480 p.
464. Wilde D. J., Sanchez-Anton J. M. Discrete optimization on a multivariable Boolean lattice // Math. Programming.— 1971.— V. 1, № 3.— P. 301—306.
465. Wilde D. J., Sanchez-Anton J. M. Multivariable monotonic optimization over multivalued logics and rectangular design lattices // Discrete Maths.— 1971.— V. 1, № 3.— P. 277—294.
466. Witzgall C. Fibonacci search with arbitrary first evaluation // Fibonacci Quart.— 1972.— V. 10, № 2.— P. 113—134.
467. Woźniakowski H. A survey of information-based complexity // J. Complexity.— 1985.— V. 1, № 1.— P. 11—44.
468. Woźniakowski H. Complexity of integration in different settings // Optimal algorithms.— Sofia: BAS, 1986.— P. 235—240.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Автоматизация проектирования 13, 261
Алгоритм 10, 22, 46, 60
— адаптивный 6, 27, 100
— асимптотически оптимальный 46, 47, 62
— бисекции 245, 246
— блочный 27, 28, 64
— вероятностный 62
— детерминированный 21, 22, 63, 65, 153, 249
— допустимый 25
— модифицированный 126, 132
— неадаптивный 26
— оптимальный 6, 30, 44, 54, 55, 58, 103, 133, 192, 201, 259, 263, 264, 265
—, — в среднем 31
—, — детерминированный 64
—, — на один шаг 47, 48, 70, 100, 113, 123, 161, 170, 191, 215, 228, 243
—, — по порядку 46, 47, 62
—, — по точности 28, 29, 36, 66, 67, 68, 85, 101, 104, 105, 136, 155, 161, 162, 163, 249, 251, 259
—, — по числу информационных вычислений 30, 47, 114
— пассивный 26, 64, 65
— повторного интегрирования 134
— последовательно - оптимальный 6, 30, 50, 56, 59, 100, 141, 161, 244
—, — по точности 61, 107, 109, 123, 167, 169, 208, 212, 219, 224, 225, 246, 254
—, — по числу информационных вычислений 61, 62, 113, 118, 123, 170, 210, 216, 219
— последовательный 6, 27, 50, 57, 69, 100
Алгоритм приближенный 125
— рандомизированный 62
— с автоматическим выбором шага 100
— — с задержкой информации 27, 64
— с ограниченным объемом памяти 27
— статистический 62
— стохастически последовательно-оптимальный 71
— стохастический 62, 249
— — пассивный 65, 66, 200, 201
— — последовательный 67, 68
— сходящийся 46
—, ε-оптимальный по точности 29, 58
—, ε-последовательно-оптимальный 61
Анализ последовательный 59

Вектор неулучшаемый 261
— результатов реализуемый 24
Внутренность множества относительная 37, 145
Вычисление алгоритмическое 23
— — допустимое 26
— информационное 23
— итоговое 23
Вычислитель 9
δ-сеть 263

Задача 11
— автоматизации проектирования 261

Задача восстановления функции 7, 10, 25, 59, 102, 153
— выпуклого программирования 81
— глобальной оптимизации 10, 31, 54, 55, 57, 191, 198
— дискретной геометрии 86
— максиминная с распадающимися переменными 256
— — со связанными переменными 256
— минимаксная 28, 256
— минимизации функции с известным значением 245
— многокритериальная 7, 244, 261
— на условный экстремум 105, 121
— наименее плотного покрытия 86
— нелинейная 110
— оптимального покрытия 86, 154, 156, 158, 198, 205, 210, 225, 227, 254, 266
— оптимизации 62
— отыскания корня монотонной функции 50
— планирования экспериментов 62
— повторного интегрирования 133
— поиска максимума 25
— поиска экстремума 31, 48, 59, 191, 198
— — унимодальной функции 48, 58, 60, 191
— приближения линейного функционала 52
— приближения оператора 29
— распределения вычислительных ресурсов 124, 219
— решения системы уравнений 10, 244, 245
— решения уравнения 47, 48, 50, 62, 244
— целочисленного программирования 143, 171
— целочисленной оптимизации 31, 110, 112, 169
— численного анализа 6, 7, 9, 10, 13, 56, 62
— — интегрирования 10, 25, 59, 62, 73, 132
— — — монотонной функции 50, 56, 118
Игра антагонистическая 63, 65, 200, 201
— — многошаговая 63, 173
— восстановления 173
— матричная 232, 235, 239
— на единичном квадрате 229, 234
— с обменом информацией 256
— с полной информацией 63, 64, 65, 173
— с полной памятью 69, 70
Измерение 23
Индикатриса длин 97
Интеграл повторный 132
Интервал локализации экстремума 48
Информация априорная 10, 11, 12, 13, 29, 31, 49
— вычислителя 10, 29
— линейная 24, 52
— нелинейная 24
— с задержкой 27
— эрмитова 124
Испытание 23

Квадратура 74
— оптимальная 95, 98
Квазиметрика 14, 16, 18, 73, 83, 200, 209, 257, 258, 262
Класс алгоритмов блочных 45
— — допустимых 21, 25
— — пассивных 26, 51, 52, 54, 55, 263
— — последовательных 27, 51, 52, 54, 55
— — с задержкой информации 45
— функций, заданный квазиметрикой 12, 14, 15, 17, 19, 20, 21, 55, 73, 133, 191, 198, 205, 256
—, — модулем непрерывности 18, 19
—, — нормой 98
— — монотонных 118, 119
Концепция оптимальности 5, 30
— — байесовская 31
— — вероятностная 31
— — информационно-статистическая 31
— — минимаксная 5, 28, 29, 47, 59, 63
— — статистическая 31

Коэффициент квадратурной формулы 74, 80
— — — оптимальный 74, 75, 82, 83, 89, 103, 152
Критерий Гросса 111
— невязки 245, 254
— осредненный 63
— оценки эффективности 5, 9, 10, 23

Лемма Фаркаша 82
ЛП$_\tau$-последовательность 87, 221

Мажоранта 76, 88, 90, 102, 119, 134, 154, 165, 205, 265
Метод выбора начальных точек 222
— — — — детерминированный 193
— — — — стохастический 193, 194
— ломаных 216
— множителей Лагранжа 121, 152
Миноранта 76, 88, 90, 102, 119, 134, 154, 165
Многогранник выпуклый 98
Множество алгоритмов блочных 28, 46
— — допустимых 25
— — пассивных 49, 104, 105, 120
— — последовательных 27, 49, 60, 104, 105, 120
— выпуклое 21, 37, 39, 52, 53, 54, 76, 77, 80, 81, 86, 145
— итоговых операций всевозможных 25, 86
— — — допустимых 25
— неопределенности 34
— Парето 261
— центрально-симметричное 21, 37, 39, 54, 86
Модель вычислений 9, 49, 70, 123, 124
— — общая 5, 6, 154, 262
— многоэкстремальной целевой функции статистическая 31
— с нефиксированным N 46
Модуль непрерывности 18

Направление допустимое 81
Невязка решения 245

Оболочка аффинная 38, 39
Оператор линейный 53, 54
— приближаемый 5
Операция итоговая 6, 23, 153
— — допустимая 25
— — линейная 36, 37, 38, 40
— — оптимальная 41
— — — при фиксированных $x_1, \ldots, x_N$ 36
— — универсально оптимальная 33, 34, 41, 104
— — универсально ε-оптимальная 33
— — фиксированная 44
— — центральная 33, 34, 88, 104, 105, 119, 135, 155, 162, 246, 251
— — ε-центральная 34
Оптимальность по Парето 261
Опыт 23
Отображение Пеано 147, 148
Оценка эффективности 29, 63
— — априорная 62
— — квадратурной формулы 80, 82

Параллелоэдр 98
Периодичность функции 20
Плотность покрытия 98
Погрешность вычислений 134, 222, 227
— — неопределенная 134, 222
— — случайная 222
— решения 10
— — максимальная на классе 29
— — относительная 49
Подход минимаксный 58
Покрытие оптимальное 91, 94, 100, 154, 155, 156, 198, 199, 260
— решетчатое 100
Понятие оптимальности 5, 9, 28, 56, 58, 63, 66
Последовательность алгоритмов асимптотически оптимальная 47

Последовательность алгоритмов, оптимальная по порядку 47
— квадратурных формул асимптотически оптимальная 96, 97
Правило множителей Лагранжа 20
— остановки 48, 61
Приближение наилучшее 9
Принцип вложенных шаров 148
— наилучшего гарантированного результата 58, 256
— оптимальности 166
Пространство линейное 9, 21, 37, 53
— метрическое 9, 34, 41, 256
Процесс вычислений 6
— квадратурный 100
— принятия решений многошаговый 165
— управляемый 6

Радиус множества чебышевский 34
— покрытия 91, 154, 198
— — оптимального 91, 154, 155, 157, 198, 254
Развертка типа кривой Пеано 147
Разность разделенная 126, 127
Рандомизация 62
— локальная 69, 70
— глобальная 69, 70
Результат, гарантированный алгоритмом 29
—, — квадратурной формулой 83
— наилучший гарантированный 42, 65, 66, 85, 104, 105, 108, 155, 163, 164, 165, 247, 254, 259
Решетка главная первого типа Вороного 100
ρ-разбиение Вороного — Дирихле 83, 84, 91
ρ-шар 90
Ряд Фурье 129

Семейство тестовое 128
Ситуация 23
— реализуемая 23, 224
Сложность информационная 29
— комбинаторная 29, 30, 31, 132, 144, 220
Совпадение налучших гарантированных результатов 49, 159, 199
Спектр оптимальной стратегии 232
Стратегия 26
— наилучшая 59
— оптимальная 64
— — смешанная 229, 233, 234, 236, 237, 239, 240
— поведения 69
— последовательно-оптимальная 64
— смешанная 65, 201
— управления 6
— чистая 65
Субдифференциал 24, 78
Схема общей модели вычислений 9
— построения последовательно-оптимального алгоритма 123, 171, 218

Тело выпуклое 93, 94, 98, 99
Теорема отделимости 39
— Цермело 64
Теория игр 69, 256
— иерархических систем 256
Тестирование алгоритмов 128, 130
Точка седловая 53, 64, 65
— — обобщенная 51
— эффективная 261
Точность абсолютная 49
— апостериорная 48, 49, 71
— априорная 71
— решения 10
— —, гарантированная алгоритмом 29
— — наилучшая гарантированная 33, 42, 43, 45
— фактическая 130

Узел квадратурной формулы 74
— — — оптимальный 74, 83, 96, 152
Упаковка оптимальная 100
Уравнение нелинейное 244

Уравнение рекуррентное 166, 173
Условие Гёльдера 17, 19, 151, 261
— Липшица 17, 19, 20, 21, 101, 102, 107, 113, 122, 123, 133, 134, 161, 164, 212, 215, 216, 242, 249, 261
— Слейтера 261
— экстремума 81

Формула квадратурная 29, 74
— — оптимальная 73, 85, 95, 96, 151
— — прямоугольников 56, 101, 103
— — трапеций 57, 120
Функционал линейный 37, 50, 52, 53, 54
— нелинейный 50
— приближаемый 5
Функция вогнутая 78, 79, 81, 82, 182, 191
— выпуклая 24, 53, 78, 79, 82, 110, 182, 191
— Лагранжа 138
Функция линейная 37
— многоэкстремальная 13, 31
— тестовая 128, 129
— универсальная наихудшая 56
— унимодальная 7, 50, 191
— целевая 31

Центр оптимального покрытия 100, 158, 209, 210
— оптимальной упаковки 100
— чебышевский 34

Шаг алгоритма 23, 59
— вычислительного процесса 23

Эксперимент 23, 26
— численный 124, 129, 131, 217
Элемент приближаемый 9, 34
ε-аппроксимация по Парето 262

Научное издание

СУХАРЕВ Алексей Григорьевич

МИНИМАКСНЫЕ АЛГОРИТМЫ
В ЗАДАЧАХ ЧИСЛЕННОГО АНАЛИЗА

Серия «Оптимизация и исследование операций»
выпуск 22

Заведующий редакцией *А. П. Баева*
Редактор *Н. И. Воронина*
Художественный редактор *Т. Н. Кольченко*
Технический редактор *И. М. Аксельрод*
Корректоры *Т. Г. Егорова, О. М. Березина*

ИБ № 32655

Сдано в набор 12.08.88. Подписано к печати 15.08.89.
Формат 84×108/32. Бумага книжно-журнальная.
Гарнитура обыкновенная. Печать высокая. Усл.
печ. л. 15,96. Усл. кр.-отт. 15,96. Уч.-изд. л. 17,53.
Тираж 4350 экз. Заказ № 311. Цена 3 р. 70 к.

Ордена Трудового Красного Знамени
издательство «Наука»
Главная редакция
физико-математической литературы
117071 Москва В-71, Ленинский проспект, 15

Четвертая типография издательства «Наука»
630077 г. Новосибирск-77, Станиславского, 25

MINIMAX ALGORITHMS IN PROBLEMS OF NUMERICAL ANALYSIS

1. AUTHOR

Aleksei G. Sukharev. Doctor of Science, Professor of Operations Research, Faculty of Computational Mathematics and Cybernetics, Moscow State University

A. Sukharev is a well-known specialist in Operations Research, Optimization and Complexity. His "Optiml Search for an Extremum" (Moscow University, 1975) is a valued research monograph while "Problems in Operations Research" (Moscow University, 1979, with coauthors), "Optimization Methods" (Nauka, 1986, with coauthors), "Operations Research, Problems and Drills" (Vysshaia Shkola, 1986, with coauthors) are among major university texts.

2. ABSTRACT

The subject of the book is complexity and optimality of algorithms in numerical analysis. Problems and algorithms are treated within a general framework based on the worst-case (minimax) concept of optimality. The book differs considerably from traditional works on optimal numerical methods. Attention to adaptive algorithms and their implementation as well as games-theoretic methods used for constructing optimal algorithms are distinctive characteristics of the book. The central role of sequentially-optimal algorithms results from the fact that the notion of sequential optimality suits a number of real computational processes better than traditional notions. Issues of program implementation, applications and numerical tests are discussed.

3. CONTENTS

Preface 5
Glossary 8

Chapter 1. **General model of computation** . 9

§ 1. Basic concepts 9
§ 2. On functional classes under consideration 12
§ 3. Classes of deterministic algorithms 21
§ 4. Minimax concept of optimality and specific notions of optimality 28
§ 5. Comparison of the best guaranteed results for adaptive and nonadaptive algorithms . . . 49
§ 6. Sequentially-optimal algorithms 56
§ 7. Stochastic algorithms 62

Chapter 2. **Numerical integration** 73

§ 1. Optimal quadratures for functional classes defined by quasimetrics 73
§ 2. Optimal quadratures for functional classes defined by continuity modules 90
§ 3. Sequentially-optimal and one-step optimal algorithms for numerical integration 100
§ 4. Numerical tests 124
§ 5. Optimal iterated numerical integration . . . 132
§ 6. Integration of multivariable functions using Peano curve 147

Chapter 3. **Recovery of functions using their values** 153

§ 1. Optimal nonadaptive algorithms 153
§ 2. Sequentially-optimal and one-step optimal algorithms for recovery of functions 161
§ 3. Solution of a multistep antagonistic game related to the problem of optimal recovery . . . 173

Chapter 4. **Search for a global extremum** 191

§ 1. On the choice of starting points for local optimization methods 192
§ 2. Optimal nonadaptive search for a functional class defined by a quasimetric 198
§ 3. Reduction of the problem of constructing a sequentialy-optimal algorithm for a functional class defined by a quasimetric to the series of problems of optimal covering 205
§ 4. Specific computational algorithms 212

§ 5. Case of approximate information 222
§ 6. One-step optimal stochastic algorithm . . . 228

Chapter 5. **Some special classes of extremal problems** 244
§ 1. Solution of equations and systems of equations 244
§ 2. Maximization of a minimum function with constrained variables 256
§ 3. Optimization with several criteria 261
Bibliography 270

Tables 5, figures 18, bibliography — 468.

СЕРИЯ «ОПТИМИЗАЦИЯ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ»

Вып. 1. А. Н. Ширяев. Статистический последовательный анализ. Оптимальные правила остановки, 1969.

Вып. 2. Б. Н. Пшеничный. Необходимые условия экстремума, 1969.

Вып. 3. Н. Н. Красовский. Игровые задачи о встрече движений, 1970.

Вып. 4. Ю. Б. Гермейер. Введение в теорию исследования операций, 1971.

Вып. 5. Н. Н. Моисеев. Численные методы в теории оптимальных систем, 1971.

Вып. 6. В. Ф. Демьянов, В. Н. Малоземов. Введение в минимакс, 1972.

Вып. 7. А. И. Пропой. Элементы теории оптимальных дискретных процессов, 1973.

Вып. 8. Н. Н. Моисеев. Элементы теории оптимальных систем, 1975.

Вып. 9. Б. Н. Пшеничный, Ю. М. Данилин. Численные методы в экстремальных задачах, 1975.

Вып. 10. Ю. М. Ермольев. Методы стохастического программирования, 1976.

Вып. 11. Ю. Б. Гермейер. Игры с непротивоположными интересами, 1976.

Вып. 12. А. М. Тер-Крикоров. Оптимальное управление и математическая экономика, 1977.

Вып. 13. Р. Г. Стронгин. Численные методы в многоэкстремальных задачах (информационно-статистические алгоритмы), 1978.

Вып. 14. Современное состояние исследования операций (сборник), 1979.

Вып. 15. В. В. Федоров. Численные методы максимина, 1979.

Вып. 16. В. Ф. Демьянов, Л. В. Васильев. Недифференцируемая оптимизация, 1981.

Вып. 17. С. А. Орловский. Проблемы принятия решений при нечеткой исходной информации, 1981.

Вып. 18. Б. Н. Пшеничный. Необходимые условия экстремума, изд. 2-е, 1982.

Вып. 19. Ю. Г. Евтушенко. Методы решения экстремальных задач и их применение в системах оптимизации, 1982.

Вып. 20. О. Хеллман. Введение в теорию оптимального поиска, 1985.

Вып. 21. Д. А. Молодцов. Устойчивость принципов оптимальности, 1987.

3 р. 70 к.