

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

САПР в электрофизике

Часть 1

Основы автоматизации проектирования

*Рекомендовано УМО «Ядерные физика и технологии»
в качестве учебного пособия
для студентов высших учебных заведений*

Москва 2011

УДК 537(075):004.9641075
ББК 22.33я7
С40

САПР в электрофизике. Ч. 1. Основы автоматизации проектирования. В 2-х ч.: *Учебное пособие* / Г.П. Аверьянов, В.А. Будкин, В.А. Воронцов, В.В. Дмитриева. М.: НИЯУ МИФИ, 2011. 164 с.

В первой части учебного пособия «САПР в электрофизике», представляющей первую часть курса аналогичного названия, изложены основы автоматизации проектирования безотносительно к конкретным приложениям. Указаны причины появления научно-технического направления и его актуальность. Представлены основные определения САПР, состав и принципы, на которых базируется эта дисциплина.

В пособии рассмотрены основополагающие положения системного анализа и ряд приложений САПР, основанных на этих положениях. Достаточно подробно рассмотрен ряд средств математического обеспечения, имеющего универсальный характер (независимо от направления применения). Это, прежде всего, методы многовариантного анализа и метод Монте-Карло. В САПР этот метод – основной при расчете допусков на внутренние параметры проектируемых изделий. Значительное внимание уделено методам нелинейного программирования – параметрической оптимизации, которая имеет обширные применения при проектировании.

Вторая часть «Методическое обеспечение» данного пособия предполагается к изданию в 2012 г.

Предназначено для студентов факультета автоматики и электроники, а также для ряда групп вечернего факультета.

Подготовлено в рамках Программы создания и развития НИЯУ МИФИ.

Рецензенты: заведующий лабораторией 27, отдел 10 «Информационно-аналитические системы радиационной безопасности человека» Федерального медицинского биофизического центра, канд. физ.-мат. наук *А.А. Игнатов*; д-р техн. наук, проф. кафедры «Системы автоматизированного проектирования» Московского государственного горного университета *А.Е. Петров*

ISBN 978-5-7262-1611-9

© Национальный исследовательский
ядерный университет «МИФИ», 2011

СОДЕРЖАНИЕ

Предисловие	5
Глава 1. Введение в проблему	7
1.1. Мотивация развития САПР	7
1.2. Основные этапы становления автоматизированного проектирования.....	15
1.3. Определение структуры и принцип построения САПР	18
1.4. Особенности САПР ЭФУ.....	21
Контрольные вопросы	26
Глава 2. Общесистемные вопросы САПР, цели и задачи проектирования	27
2.1. Краткие сведения о системном анализе	27
2.2. Блочно-иерархический подход – основной метод исследования сложных систем.....	30
2.3. Жизненный цикл сложной технической системы (ГОСТ 22487–77).....	32
2.4. Формализация процесса проектирования	35
Контрольные вопросы	38
Глава 3. Математическое обеспечение САПР (методы многовариантного анализа)	39
Введение	39
3.1. О моделях и моделировании.....	40
3.2. Математическое моделирование.....	43
3.3. О методах многовариантного анализа.....	49
3.4. Метод статистических испытаний (метод Монте-Карло).....	53
Контрольные вопросы	84
Глава 4. Математическое обеспечение САПР – решение задачи синтеза, методы оптимизации	85
4.1. Введение в проблему параметрического синтеза.....	85
4.1.1. Общие сведения, постановка задачи.....	85
4.1.2. Выбор целевой функции	91
4.1.3. Задачи нелинейного программирования	96
4.2. Методы поиска минимума одномерных унимодальных функций.....	101

4.2.1.	Методы исключения интервалов (дихотомия, деление отрезка пополам, золото сечение, метод Фибоначчи).....	101
4.2.2.	Полиномиальная аппроксимация (метод с использованием квадратичной аппроксимации, методы с использованием производных, метод поиска с использованием кубической аппроксимации).....	112
4.3.	Поиск минимума функций многих переменных	117
4.3.1.	Методы прямого поиска (симплекс-метод, методы спуска, метод покоординатного спуска, метод Хука-Дживса).....	123
4.3.2.	Метод сопряженных направлений. Метод сопряженных направлений Пауэла	136
4.3.3.	Градиентные методы (метод Коши, метод Ньютона).....	139
4.3.4.	Методы сопряженных градиентов (алгоритм метода сопряженных градиентов, алгоритм Флетчера-Ривса, алгоритм Дэвидона-Флетчера-Пауэла (с переменной метрикой))	144
4.4.	Методы случайного поиска (стохастическая аппроксимация).....	149
4.5.	Поиск глобального минимума целевой функции	150
4.5.1.	Поиск глобального минимума функции одной переменной (метод ломаных, метод покрытий)	150
4.5.2.	Поиск глобального минимума функции многих переменных	153
4.6.	Методы оптимизации в задачах с ограничениями	156
	Контрольные вопросы	159
	Список литературы	161

ПРЕДИСЛОВИЕ

Автоматизированное проектирование представляет одно из основных направлений научно-технического прогресса конца XX в. Оно не связано с какими-то фундаментальными открытиями или новыми достижениями именно в этой области, а целиком и полностью обязано колоссальным технологическим достижениям в области новых информационных технологий. Развитие этого направления показало, что не всегда даже очень заманчивые идеи (такие, как высокотемпературная сверхпроводимость или термоядерный синтез) могут привести к практическим результатам, а вдумчивое осознание современных технологических возможностей позволяет решить важнейшие проблемы современности.

Всеобщий интерес к автоматизации проектирования и интенсивное его развитие началось с 60-х годов прошлого столетия и связано с двумя основными факторами: значительным усложнением проектируемых объектов, приведшим к кризису в традиционных методах проектирования, и революционными событиями, происходящими в электронике и вычислительной технике. Именно в этот период очень своевременно сформировалась новая база, которая позволила преодолеть критическую ситуацию, сложившуюся к тому времени в проектировании.

Под автоматизацией проектирования понимают систематическое применение средств вычислительной техники в процессе проектирования при научно обоснованном распределении функций между проектировщиком и компьютером и при научно обоснованном выборе метода машинного решения задач. В настоящее время общепринятой наилучшей организацией считается применение систем автоматизированного проектирования (САПР), основными компонентами которых являются технические средства, математическое, общее и специальное программное, информационное и организационное обеспечения.

Необходимо отметить, что САПР – специализированная система. САПР электроники значительно отличается от САПР радиотехники приборостроения, а тем более от САПР машиностроения и таких узкоспециализированных систем, как, например, разработка пульта управления пилота самолета.

Отмечая специфичность САПР конкретных предметных областей, необходимо подчеркнуть, что для многих отраслей техники ряд основополагающих положений теории и практики САПР имеет достаточно общий характер, например принципы построения общего и специального программного обеспечения, формирование типовых последовательностей задач и программ, организация взаимодействия проектировщика и компьютера, подходы к построению математических моделей проектируемых объектов и т.д. Универсальность многих положений автоматизации проектирования имеет ту же природу, что и общность приемов математического исследования различных физических объектов и явлений. Поэтому развитие автоматизированного проектирования выражается, прежде всего, в совершенствовании и углублении именно математических приемов исследования.

Первая часть учебного пособия «САПР в электрофизике» посвящена общим вопросам автоматизации проектирования, независимым от конкретных приложений. В первой главе даны мотивации САПР и основные определения, состав и принцип создания. Во второй главе рассматриваются общесистемные вопросы САПР, необходимые при изучении и проектировании сложных технических систем. В третьей главе приведены математические методы решения задачи синтеза и методы многовариантного анализа.

ГЛАВА 1. ВВЕДЕНИЕ В ПРОБЛЕМУ

1.1. Мотивация развития САПР

Дисциплина САПР, так же как аналогичное научное направление, возникла сравнительно недавно: в 60–70-х годах прошлого столетия (и тысячелетия), но тем не менее сравнительно быстро нашла признание и успешно внедряется в практику во всех развитых странах мира, хотя следует отметить, что ввиду новизны этой проблемы и разнообразных сфер ее внедрения представление о ней не всегда однозначны даже среди специалистов. В зарубежной практике это направление определяется как CAD-Computer Aided Design (разработка с помощью компьютера), во Франции принято сокращение CAO [2]. Эти сокращения часто применяются для обозначения пакетов прикладных программ, используемых разработчиками в различных аспектах научно-технической деятельности (Auto CAD, P-CAD, Mat CAD и т.п.). Разработки крупных программных комплексов связаны с так называемой CASE-технологией (Computer Aided Software Engineering). Все эти направления можно рассматривать как составные части общей проблемы автоматизированного проектирования.

Чтобы понять основные причины интенсивного развития автоматизированного проектирования, необходимо определить задачи, решаемые в процессе проектирования.

Как правило, под проектированием (на первый взгляд) понимается изготовление чертежно-конструкторской документации (кульман, карандаш, калька и т.п.). Безусловно, изготовление чертежей – трудоемкий, рутинный и длительный процесс, но он является лишь завершающим этапом проектирования. Хотя автоматизация проектирования начиналась со стремления кардинально изменить процедуру изготовления чертежно-конструкторской документации. Однако следует отметить, что существует понятие «функциональное проектирование», составляющее основу проек-

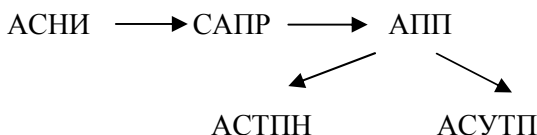
тирования. Оно включает в себя решение трудоемких задач, связанных с определением принципов построения объектов проектирования и оценки свойств на основе исследования процессов их функционирования. Автоматизация функционального проектирования предполагает решение этих задач с помощью функциональных математических моделей, объектов проектирования на микро-, макро- и метауровнях. Именно об этом проектировании, в основном, будет идти речь в дальнейшем. Вопросы машинной графики, машинной геометрии и т.п. рассмотрены в смежных разделах.

В самом общем виде под автоматизацией проектирования можно понимать технологию использования вычислительной техники для оказания помощи проектировщику при выработке, модификации, анализе и оптимизации проектных решений, как говорил Норберт Винер: «отдайте человеку человеческое, а вычислительной машине машинное. В этом и должна, по-видимому, заключаться разумная линия поведения при организации совместных действий людей и машин».

Существует два научных направления смежных с САПР и также базирующихся на использовании различных средств вычислительной техники. Начальный этап разработки новых изделий и поиска новых принципов связан с так называемыми АСНИ – автоматизированными системами для научных исследований. Эта стадия научного поиска существовала всегда и являлась лидером в деле использования автоматизированных систем для изучения новых идей и использования новых принципов. АСНИ предваряет непосредственное проектирование, являясь более универсальным этапом, поскольку значимые результаты этой деятельности могут быть использованы при проектировании в различных областях техники, являясь составной и очень важной на современном этапе частью научного поиска. В современном понимании это связывают с модным словом инновации (раньше использовалось слово внедрение). С другой стороны, на завершающем этапе процесса создания новых изделий, также связанным с интенсивным (в современных условиях) использованием вычислительной техники, находятся так называемые системы АПП – автоматизации производственных процессов. Они включают две функционально-различных и взаимодополняющих системы АСТПП (автоматизированные системы

технологической подготовки производства) и АСУТП (автоматизированные системы управления технологией производства).

Таким образом, весь цикл создания новых изделий может быть представлен в виде последовательности следующих этапов, в основе технологии каждого из которых находятся средства вычислительной техники:



В американской литературе, в связи с вышесказанным, употребляется сокращение CAD/CAM, где CAM – Computer Aided Manufacturing. Это лишний раз указывает, что использование средств вычислительной техники на всех этапах от научного поиска до изготовления готовых изделий является основной тенденцией научно-технического прогресса на современном этапе.

Лидеры подобного подхода – разработчики электронной техники, точнее микроэлектронной техники, где разработка и изготовление схем высокой степени интеграции представляет замкнутый цикл на основе автоматизированных систем от разработки принципиальных схем, их моделирования и оптимизации, разработки конструкции до технологической подготовки и изготовлении готовых изделий.

Выделение автоматизированного проектирования в отдельную дисциплину, как самостоятельного научно-технического направления, связано со значительными изменениями, произошедшими при подходе к технологии проектирования, значительному усложнению этой технологии в связи с появлением широкого спектра новейших технологических средств, математического и программного обеспечения, ориентированного, в том числе и на изготовление чертежно-конструктивной документации.

Предмет автоматизации проектирования на современном этапе включает следующее:

- формализацию проектных процедур;
- структурирование и формализацию процессов проектирования;
- разработку моделей проектируемых объектов;
- разработку методов и алгоритмов проектных задач;

- способы построения технических средств;
- разработку входных языков проектирования и т.п.

Говоря о САПР, следует отметить, что, с одной стороны, это узкоспециализированная система. Так, САПР электроники отличается от САПР радиотехники и приборостроения, не говоря уже о САПР машиностроения и такой узкоспециализированной системы автоматизированного проектирования, как разработка пульта управления пилота самолета. Очень специфично проектирование ядерно-энергетических систем и электрофизических установок.

С другой стороны, отмечая специфику автоматизации проектирования в различных областях техники, необходимо подчеркнуть следующее. Ряд основополагающих положений теории и практики САПР носит универсальный характер, это связано, прежде всего, с тем, что основу любой системы автоматизации проектирования составляют средства вычислительной техники, являющиеся универсальным инструментом и требующие подробного изучения. Проблемы использования и принципы построения технических средств, структура программного обеспечения как универсального, так и специализированного, выбор инструментальных средств разработки прикладных программ, принципы разработки математической модели входных языков носят в значительной степени универсальный характер и практически не зависят от области применения.

В связи с этим в изучении автоматизации проектирования можно выделить два уровня:

- изучение общих, универсальных составляющих САПР, не зависящих от области применения;
- изучение специализированных средств, отражающих специфику конкретной предметной области.

Всеобщий интерес к автоматизации проектирования и интенсивное его развитие начинается с 60-х годов прошлого столетия и связан с двумя основными факторами: кризисом в традиционных методах проектирования и революционными событиями, происходящими в электронике и вычислительной технике. Именно в этот период очень своевременно сформировалась новая база, которая позволила преодолеть критическую ситуацию, сложившуюся к тому времени в проектировании.

Следует отметить, что, хотя автоматизированное проектирование и представляет одно из основных направлений научно-

технического прогресса конца XX в., оно не связано с какими-то фундаментальными открытиями или новыми достижениями именно в этой области, а целиком и полностью обязано колоссальным технологическим достижениям в смежной области – развития новых информационных технологий. Развитие этого направления показало, что не всегда даже очень заманчивые идеи (например, сверхпроводимость или термоядерный синтез) могут привести к положительным практическим результатам, а вдумчивое осознание современных технологических возможностей позволяет решить важнейшие проблемы современности.

Как указывалось ранее, первая (и основная) причина появления САПР связана с кризисом в проектировании – из-за несоответствия традиционных методов проектирования новым потребностям проектировщиков, постоянно усложняющимся объектами проектирования, появления понятия сложной технической системы (СТС). Так, в среднем по сравнению с 50-ми годами, сложность объектов проектирования в 60–70-х годах увеличилась в 5–6 раз. Для сравнения, документация на современный самолет того времени по весу была примерно равна весу самого самолета. Очень серьезные изменения, потребовавшие совершенно новых подходов к проектированию, происходили в электронике, что связано с колоссальным увеличением степени интеграции интегральных микросхем – ИМС (переход от отдельных транзисторов 50-х годов к ста и более миллионов транзисторов в одной ИМС) и появлением микропроцессоров, степень интеграции в которых приближается к одному миллиарду.

Характерный пример этой тенденции был связан с разработками кафедры ЭФУ МИФИ. В начале 70-х годов прошлого столетия кафедра на хозяйственной основе разрабатывала и устанавливала для различных организаций небольшие линейные ускорители.

Для оснащения ускорителей необходимым вспомогательным оборудованием – электронными системами управления, пультами операторов и придания установкам современного дизайна, была предпринята попытка привлечь для решения этих задач разработчиков из ГДР. Однако после составления совместного технико-экономического обоснования выяснилось, что стоимость вспомогательного оборудования втрое превышает стоимость основной части установки самого ускорителя, и совместное предприятие не состоялось.

Ситуация с усложнением объектов проектирования и появление СТС привела к ряду проблем, требующих оперативного решения.

Значительно увеличились сроки проектирования. Так, составление проекта нового торгового судна требовало 2–4 года, самолета – 4–6 лет, ЭВМ – 3–4 года. Создание крупного ускорителя – от 4 до 10 лет. Причем подготовка предварительных соображений (предпроектные исследования очень важный этап проектирования) занимала от двух до четырех лет.

Увеличение времени, затрачиваемого на проектирование, само по себе нежелательно, так как значительно растут сроки ввода новых изделий в эксплуатацию, более того, в ряде областей техники (в частности, в электронике) было невозможно традиционными методами создавать новую продукцию. Как следствие увеличения сроков проектирования является и то, что идеи, заложенные в проекте, за время его разработки устаревают еще до ввода в эксплуатацию новых изделий, а это связано с опережающими темпами роста научно-технического прогресса. Если в 50-е годы прошлого столетия жизненный цикл эксплуатации разработок равнялся профессиональной жизни разработчика-человека, то к 60–70 годам время разработки, зачастую, превышало время эксплуатации и к моменту создания (выпуска) изделий они оказывались устаревшими. Типичным примером такой ситуации явилась выпускаемая в Советском Союзе единая система вычислительных машин – ЕС ЭВМ. Таким образом, одновременно с увеличением сроков проектирования происходит сокращение времени их жизни.

В качестве примера приводим сроки сооружения крупнейших ускорителей того времени:

Ускоритель	Годы создания
Синхрофазотрон 10 ГэВ (Дубна)	1952–1957
Проточный синхротрон 26 ГэВ (CPS, ЦЕРН)	1955–1959
Электронный синхротрон 1,3 ГэВ (Томск)	1958–1965
Электронный синхротрон 7,2 ГэВ (DESY)	1959–1964
Проточный синхротрон 76 ГэВ (Протвино)	1961–1967
Линейный ускоритель электронов 22 ГэВ (Стэнфарт, США)	1962–1966
Мезонная фабрика 75–100 МэВ, 100–800 МэВ (Лос-Алиное, США)	1962–1972
Проточный синхротрон 200–500 ГэВ (Батавия, США)	1968–1972
Супер протонный ускоритель SPS, 400 ГэВ (ЦЕРН)	1970–1976

Еще одним фактором, приведшим к кризису проектирования в связи с появлением СТС, явилось значительное увеличение лиц,

занятых в разработке проектов (конструкторов, чертежников, копировщиков и т.п.), и невысокая престижность конструкторского труда, ввиду рутинного характера большинства проектных процедур. Что, в свою очередь, приводило к значительному удорожанию проектов.

Вследствие этих причин в цикле создания новой техники «научное исследование – проектирование – производство» узким местом явилось не отсутствие научных достижений и даже не изготовление, а медленное освоение имеющихся научных достижений через проектирование.

Вторая причина, связанная с появлением САПР, позволившая преодолеть кризисные явления – это развитие новой технологической базы проектирования в связи с появлением и бурным развитием новейших информационных технологий на базе средств вычислительной техники.

Чтобы понять суть перемен, произошедших в проектировании, и место новых информационных технологий в этом процессе, необходимо рассмотреть детально виды деятельности, присутствующие в этом процессе. На рис. 1.1 представлен набор разновидностей этой деятельности и взаимосвязь между ними. В нижней части рисунка представлены виды работ, необходимые для выполнения задачи проектирования и их процентное соотношение. Из этого рисунка следует, что весь процесс проектирования представляется как чередование творческих и формальных (рутинных) видов деятельности, причем последние составляют большую его часть (> 90 %). К ним относятся – расчеты, хранение, поиск, обработка информации и результатов экспериментов, а также изготовление технической документации.

Большой ассортимент технических средств современной вычислительной техники, развитие программного обеспечения, автоматизация программирования, увеличивающиеся ресурсы вычислительных систем, интеллектуальные интерфейсы и т.п. предоставляют возможность проектировщикам перенести на компьютер рутинные виды деятельности, значительно повысив эффективность проектирования.

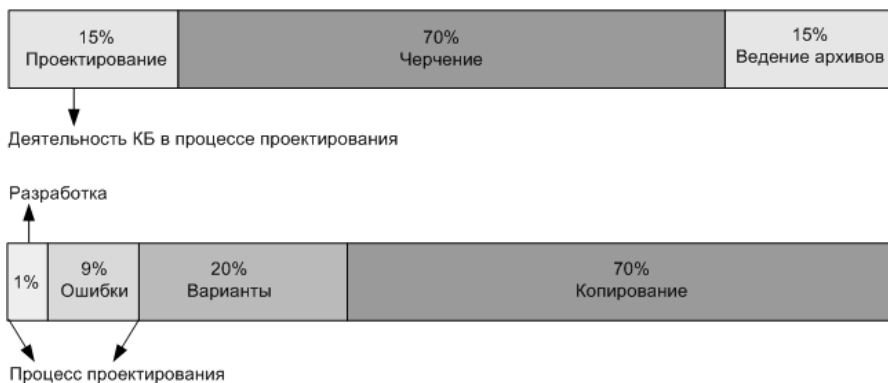


Рис. 1.1. Различные виды деятельности в конструкторском бюро (КБ)

1.2. Основные этапы становления автоматизированного проектирования

Таким образом, автоматизация проектирования с помощью средств вычислительной техники – основной путь, на котором преодолеваются противоречивые трудности, связанные с сокращением сроков проектирования при одновременном увеличении сложности разработок.

Цель автоматизированного проектирования – объединение конструктора и ЭВМ в единую команду, способную приходить к поставленным целям в задачах проектирования более эффективно, чем каждый из них работающий по отдельности. Принципы, лежащие в основе этого разделения, находятся в зависимости от последних достижений в области новых информационных технологий.

Основные задачи, которые ставятся при переходе на автоматизированное проектирование, следующие:

- сократить сроки проектирования;
- снизить материальные затраты на проектирование;
- повысить качество проектирования;
- сократить количество конструкторов и чертежников, занятых на рутинных операциях.

Хотя автоматизация проектирования – основной путь развития проектирования, но этот путь не тривиален и не всегда все поставленные цели достигаются одновременно:

- на смену кульману приходят достаточно дорогие технические средства, мощные рабочие станции, устройства графического ввода-вывода чертежей, соответствующее программное обеспечение и многое другое;
- математическая постановка задач для большинства проектных процедур неочевидна, а их последующая алгоритмизация требует разработки оригинальных методов, что в значительной мере определяет содержание теории САПР.

Перечисление проблем, возникающих при внедрении САПР, можно продолжить, но, несмотря на эти трудности, успешное внедрение САПР в практику проектирования подтверждает реальность решения перечисленных проблем.

Уже на ранних стадиях развития новых методов проектирования успешно решались эти задачи в станкостроении, электронной тех-

ники, судостроении и т.п. Впечатляющие достижения демонстрируют зарубежные фирмы. Так, фирма Boeing способна в течение года запустить в производство новый авиалайнер (к примеру, отечественный авиалайнер ТУ 234 запускался в производство в течение пяти лет).

Можно выделить несколько последовательных этапов развития САПР, тесно связанных с динамикой развития средств вычислительной техники и методов ее использования.

Начальный этап связан с использованием вычислительной машины при решении сложных математических задач, возникающих при проектировании. На этом этапе важнейшую роль играло развитие методов вычислительной математики и ее внедрение в инженерную практику.

Развитие методов проектирования тесно связано с развитием методики использования вычислительной техники. Решение инженерных математических задач на компьютере (так же как и решение любых других задач) строится по следующей схеме.

1. Математическая формулировка задачи.
2. Разработка алгоритма.
3. Выбор численных методов решения задачи.
4. Кодирование алгоритма (программирование).
5. Запись программы на различные носители устройств ввода-вывода компьютера.
6. Отладка программы и ее тестирование.
7. Решение задачи.
8. Обработка результатов.

На начальном этапе использования компьютеров только седьмой-пункт выполнялся без участия человека (оператора или программиста), а эффективность использования компьютеров была крайне низкой.

Решающую роль в переходе на новый уровень использования компьютеров, имеющий важное значение в автоматизации проектирования, сыграла тенденция к универсализации задач и разработка единых подходов к целому классу расчетно-проектных процедур. На начальном этапе этот подход связан с реализацией модульного программирования, созданием библиотек по специальности в виде подпрограмм функций и процедур как на системном уровне, так и в виде внешних процедур, собираемых на стадии компоновки.

Были разработаны библиотеки как для широкого круга применений, например библиотеки научных подпрограмм на Фортране фирмы IBM, так и более специализированные библиотеки для различных научно-технических приложений.

Основная задача, которая ставилась на этом этапе, по возможности избавить разработчика от рутины алгоритмизации и кодирования задачи, т.е. исключить ручной труд на этапах 2–5.

Математическая постановка задачи, безусловно, определяется разработчиком. Широкая универсализация задач, позволяющая абстрагироваться от деталей алгоритмизации и программирования (кодирования), увеличивает накладные расходы на ресурсы компьютеров, что при современных темпах развития технических средств не актуально. Однако большинство универсальных библиотек не учитывали особенности конкретных приложений и в ряде случаев были мало пригодны на практике.

На этом же этапе автоматизации проектирования успешно решались задачи новых способов оформления технической документации, оперативной обработки и отображения результатов проектирования. Создавались и быстро совершенствовались специализированные пассивные и интерактивные устройства и системы машинной графики – графопостроители, дигитайзеры, разнообразные дисплеи и т.п. Одновременно развивалось специализированное программное обеспечение машинной графики и ее более высокая ступень – машинная геометрия. Лидерами этого направления были ведущие фирмы США. Так, уже в 1963 г. сотрудники МТИ (Массачусетского технологического института) представили доклад, в котором демонстрировали возможности формирования изображения на экране и манипулирование им в реальном масштабе времени, что явилось началом интерактивной машинной графики (ИМГ). И уже к концу 60-х годов (прошлого столетия) крупнейшие американские концерны (General Motors, IBM, Lockheed, MC Donal Douglas) включились в создание ИМГ. К началу 70-х годов появилось несколько поставщиков САПР/АПП.

На этом этапе развитие автоматизации проектирования виделось в дальнейшем увеличении мощности компьютеров и развитии специализированной периферии. Однако довольно быстро выяснились ограниченные возможности этого подхода при проектировании сложных технических систем, требующих системного подхода.

Как выразился известный специалист в области информационных технологий академик Н.Н. Моисеев: «Простое наращивание вычислительной мощности без качественного изменения всей ее информационной основы, без надлежащей организации человеческого труда не может дать ожидаемого эффекта, а напротив приводит к "вавилонскому столпотворению"».

Наглядный пример «вавилонского столпотворения» – ситуация, сложившаяся на американской фирме Boeing. В 1970 г. на фирме был проведен тщательный анализ результатов около 700 расчетных работ с использованием ЭВМ. Результаты были ошеломляющими. Каждое второе решение расчетной задачи было неверным. В большинстве случаев это не было обнаружено своевременно и результаты пошли в качестве исходных данных в решение других задач. Дополнительные затраты на устранение каждой ошибки составляли от 2-х до 15 человекомесяцев.

Как показал анализ причин ошибок, они были связаны, отнюдь, не с низким качеством компьютерных программ, а с низким качеством технической культуры использования вычислительной техники в проектировании – отсутствия четкой организации вычислений, из-за путаницы в передаче информации, из-за ошибок, внесенных в результаты самими исполнителями. Именно подобного сорта обстоятельства привели к необходимости системного подхода к автоматизации проектирования, т.е. к переходу на новый этап систем автоматизированного проектирования.

1.3. Определение структуры и принцип построения САПР

По существу, современная САПР представляет систему проектирования, в которой произведено рациональное, на данном этапе развития технических и программных средств ЭВМ, распределение функций между людьми (проектировщиками) и средствами вычислительной техники. При этом вся процедура проектирования рассматривается с системных позиций как единый процесс, начинающийся с разработки технического задания и заканчивающийся изготовлением технической документации, подготовленной для запуска на изготовление изделия в производство. ГОСТ вводит следующее определение САПР [1]: «САПР – это комплекс средств АПР (автоматизированного проектирования), взаимосвязанных с

необходимыми подразделениями проектной организации или коллективом специалистов, выполняющих АПР».

По своему назначению основные подсистемы, входящие в структуру САПР, могут подразделяться на:

- проектирующие, имеющие объектную ориентацию и реализующие определенный этап (стадию) проектирования;
- обслуживающие, имеющие общесистемное применение и обеспечивающие поддержку функционирования проектирующих систем, а также оформление, передачу и вывод полученных результатов.

Средства автоматизации проектирования можно сгруппировать по видам обеспечения, которое включает в себя следующее.

Математическое обеспечение (МО), основа которого – алгоритмы, по которым разрабатывается программное обеспечение. Элементы МО в САПР чрезвычайно разнообразны. Среди них – принципы построения функциональных моделей, методы решения алгебраических и дифференциальных уравнений и т.п.

Разработка МО – самый сложный этап создания САПР. По назначению и способам реализации МО САПР делится на две части:

- математические методы и построенные на их основе математические модели, описывающие объекты проектирования;
- формализованное описание технологии автоматизированного проектирования.

Программное обеспечение (ПО) – совокупность всех программ и эксплуатационной документации к ним, необходимых для выполнения автоматизированного проектирования.

Имеется две части ПО:

- общесистемное ПО, предназначенное для организации функционирования технических средств, т.е. для планирования и управления вычислительным процессом;
- специальное (прикладное ПО) математическое обеспечение, ориентированное на выполнения проектных задач.

В настоящее время разработано большое количество пакетов прикладных программ (ППП), реализующих САПР в разнообразных областях науки и техники: в электронике, машиностроении, математике и т.п.

Информационное обеспечение (ИО) САПР, основу которого составляют данные, которыми пользуется проектировщик в процессе

проектирования. Они могут быть представлены в виде документации на различных носителях, содержащих сведения справочного характера.

Различают следующие способы ведения информационного фонда САПР: использование файловых систем и использование баз, банков данных, создание информационных программ адаптеров.

Техническое обеспечение – совокупность взаимосвязанных и взаимодействующих технических средств, предназначенных для выполнения автоматизированного проектирования.

Лингвистическое обеспечение – специальные языковые средства (языки проектирования), предназначенные для описания процедур автоматизированного проектирования и проектных решений – входные языки ППП. Сюда же следует отнести традиционные инструментальные средства операционных систем – традиционные (императивные) языки программирования.

Методическое обеспечение включает входящие в САПР документы, регламентирующие порядок ее эксплуатации, включая описание ППП.

Организационное обеспечение – инструкции, приказы, штатное расписание, квалификационные требования и другие документы, регламентирующие организационную структуру подразделений проектной организации.

Основные принципы построения САПР.

САПР – это человеко-машинная система. Тесное взаимодействие человека и ЭВМ в процессе проектирования – один из принципов построения и эксплуатации САПР. В настоящее время и, по крайней мере, в ближайшие годы создание САПР «не угрожает» монополии человека при принятии узловых решений в процессе проектирования. Человек должен решать в САПР задачи, формализация которых не достигнута и которые решаются на основе опыта и интуиции человека.

САПР – иерархическая система, она реализует комплексный подход к автоматизации всех уровней проектирования. Следует особо подчеркнуть целесообразность комплексного характера САПР, так как автоматизация проектирования на одном из уровней при сохранении старых форм проектирования не дает заметного суммарного эффекта.

САПР – совокупность информационно согласованных подсистем, что означает – все или большинство последовательностей задач проектирования обслуживается информационно согласованными программами. Две программы называются информационно согласованными, если все те данные, которые представляют собой объект переработки в обеих программах, входят в числовые массивы, не требующие изменений при переходе от одной программы к другой. Плохая информационная согласованность превращает САПР в совокупность хаотических программ.

САПР – открытая и развивающаяся система, так как имеется как минимум две причины, по которым она меняется во времени:

- разработка такой сложной системы, как САПР, занимает продолжительное время, и экономически выгодно вводить отдельные ее части по мере готовности;
- прогресс в вычислительной технике и развитие вычислительных методов приводит к накоплению более совершенных методов автоматизации проектирования и необходимости замены старых.

В связи с этим открытость системы предполагает замену отдельных блоков системы без глобальной перделки системы в целом.

САПР – специализированная система с максимальным использованием унифицированных модулей. Требование высокой эффективности и универсальности, как правило, противоречивы. Высокой эффективности достигают за счет специализации. При этом число САПР растет, растет и их стоимость. Чтобы снизить стоимость, целесообразно включать в их состав недорогие унифицированные модули.

1.4. Особенности САПР электрофизических установок

Как уже указывалось, общие базовые принципы разработки автоматизированных систем проектирования и традиционный набор видов обеспечения, использующих достижения современных (на данном этапе развития) информационных технологий, каждое направление проектирования разнообразных технических устройств имеет свои характерные и весьма значительные особенности. Эти особенности связаны, прежде всего, с особенностями математического, программного, лингвистического, информационного и т.п. видов обеспечения, отражающих специфику предметной области и

типов разрабатываемых изделий. Специфика конкретной предметной области разработки обычно отражается в операционной среде ППП автоматизированного проектирования. Например, на САПР электроники ориентированы такие широко известные пакеты, как P-CAD, OrCAD, MicroCap, Electronics Work Bench и т.п.

Большую популярность и массовое распространение получили ППП AutoCAD фирмы Autodesk RF. Эта – мощная операционная среда, включающая в себя большое количество общеинженерных приложений, ориентированных, в основном, на машиностроение, архитектуру и строительство. Существует большое количество ППП, ориентированных на определенные более узкие области техники – САПР в авиастроении, кораблестроении и т.п.

В большинстве проектных организаций общеинженерного профиля значительная часть времени тратится на изготовление чертежно-конструкторской документации, т.е. на черчение. В связи с этим именно эта часть работы изначально понималась проектированием и требовала разработки и внедрения новых технологий с использованием машинной графики, машинной геометрии, создания специализированных пассивных и интерактивных устройств ввода-вывода, систем машинной графики и автоматизированных рабочих мест (АРМ) конструктора.

Разработка ПО для поддержки этих устройств явилась необходимым атрибутом для создания в дальнейшем ППП автоматизации проектирования (CAD).

Методы расчета в большинстве общеинженерных дисциплин хорошо изучены, стандартизированы, их результаты, как правило, представлены в виде таблиц, номограмм и т.п. Так, в вакуумной технике основные элементы вакуумных систем определены стандартами. Выбор типа вакуумных прокладок, размеры фланцев и необходимое количество болтов и их размеры для обеспечения надежного уплотнения определяются из таблиц и нет потребности ни в прочностных и иных расчетах.

Свои характерные особенности имеет проектирование в электронике (микроэлектронике) и электротехнике, где основными расчетными соотношениями являются компонентные законы Ома и топологические уравнения Кирхгофа, а конструирование, в основном, связано с оптимизацией расположения элементов с целью уменьшения длины соединяющих проводников и проблемы теплоотвода.

Как правило, ранее перечисленные изделия выпускаются массово (крупными сериями), а жесткие требования стандартов связаны с жесткими требованиями к разработке конструкторской документации.

Особенности автоматизации проектирования в электрофизике связаны с особенностями этого класса установок.

С самого начала разработка ускорителей и других установок электрофизики потребовала проведения сложных вычислительных работ большого объема с привлечением мощных вычислительных систем. С другой стороны, не существовало как таковой промышленности по производству подобных установок. Разработки, как правило, проводились (и проводятся) физическими исследовательскими институтами, хотя отдельные типы ускорителей для промышленности и медицины, которые получили достаточно широкое распространение, производились небольшими сериями рядом зарубежных фирм. Общепринятых стандартов на эти установки пока не существует. Среди наиболее характерных особенностей этих установок, с точки зрения конструкторской разработки, можно отметить следующие.

Мелкосерийность – для промышленности и медицины ускорители выпускаются некоторыми фирмами в единичных экземплярах, или небольшими сериями (в пределах десятков штук).

Крупные исследовательские ускорительные комплексы уникальны и неповторимы. Это, с одной стороны, объясняет их высокую стоимость, а с другой – отсутствие в необходимости стандартизации.

Очень большое количество принципов ускорения и, как следствие, большое количество типов ускорителей. Это электростатические, линейные, индукционные мощные импульсные на основе генераторов Аркадьева-Маркса, линейные высокочастотные и самые разнообразные циклические ускорители. В них ускоряются самые разнообразные частицы на различные энергии для различных целей.

Сложность теории ускорителей. Каждый тип ускорителей предусматривает свою методику расчета и требует серьезной научно-исследовательской проработки и большого объема компьютерного численного моделирования. Важнейшее значение при проектировании электрофизических установок (ЭФУ) связано с решением задач синтеза – выбора оптимальных вариантов на различных

уровнях проектирования: от выбора типа установки и круга решаемых на ней задач до оптимизации отдельных компонент выбранной установки.

Большое количество общинженерных систем. Это вакуумная система, система компоновки, установки и юстировки, система автоматизации измерений и управления, энергетическая система и т.п.

Проектирование этих подсистем не имеет непосредственного отношения к САПР ЭФУ и решается в смежных областях техники, имеющих свои системы автоматизированного проектирования.

Однако наряду с перечисленными общинженерными подсистемами, проектирование которых достаточно хорошо проработано и существуют соответствующие системы автоматизированного проектирования, имеются четыре подсистемы, характерные только для ЭФУ – это ускоряющая, фокусирующая, инжектирующая подсистемы и каналы транспортировки заряженных пучков. Эти подсистемы представляют основы САПР ЭФУ.

Проектирование этих подсистем связано с выбором и оптимизацией как самых ускоряющих и фокусирующих подсистем, так и их параметров. При анализе этих структур приходится решать граничные задачи электромагнитостатики и электродинамики резонансных волноводных ускоряющих структур. Показателем эффективности разрабатываемых подсистем ЭФУ является параметры ускоряемого выходного тока заряженных частиц, которые определяются из решения уравнений движения частиц в ускоряющих и фокусирующих электромагнитных полях. Особую сложность представляют так называемые самосогласованные задачи, в которых ускоряемый ток и заряд частиц оказывают заметное влияние на ускоряющие и фокусирующие поля.

В связи с этим под САПР ЭФУ (или САПР ускорителей заряженных частиц) будем понимать стадию предпроектных исследований, связанную с разработкой математических моделей, их анализом и синтезом для решения вышеперечисленных задач, т.е. разработку ускоряющих, фокусирующих и транспортирующих систем пучков заряженных частиц.

Разработка компьютерных программ для решения перечисленных задач была начата с конца 50-х годов прошлого века во всех крупных ускорительных центрах мира. Эти работы были, как правило, связаны с созданием национальных ускорительных центров.

Однако ориентация этих программ на определенные технические средства, операционные среды и различные версии языков программирования того времени затрудняли переместимость программных продуктов и использования их в других центрах. В дальнейшем, в связи со стандартизацией инструментального ПО и унификаций операционных систем, эти трудности были преодолены. Наиболее известным примером ускорительного САПР является разработанный в Европейском центре ядерных исследований (ЦЕРН, Швейцария) фундаментальный программный комплекс MAD, получивший широкое распространение в мире, подробно и качественно документированный, однако для разработки небольших установок, которые используются в прикладных задачах для самых разных целей в промышленности, медицине, сельском хозяйстве и т.п. ППП MAD является, с одной стороны, избыточным, а с другой – не всегда удовлетворяет специфическим требованиям различных установок.

В связи с этим в настоящее время в практике проектирования ускорителей используются узкоспециализированные ППП, о которых будет сказано в дальнейшем.

Фактически все разработанные и разрабатываемые программные продукты при проектировании небольших установок прикладного характера ориентированы на две задачи:

- расчет электромагнитных полей в ускоряющих и фокусирующих системах, а также программы расчета тепловых полей в этих системах (эти программы имеют обширную сферу применения, выходящую за пределы электрофизики и разрабатываются специалистами из смежных областей);
- расчет динамики ускоряемых частиц в выбранных структурах (и оптимизация структур через анализ динамики заряженных частиц), при необходимости учитывается влияние электромагнитного поля, создаваемого пучком на ускоряющее поле (это, как правило, узкоспециализированные программы, ориентированные на определенный класс установок).

Контрольные вопросы

1. Что понимается под фундаментальным проектированием и какие задачи имеются на этом уровне?
2. Перечислите основные причины кризиса традиционного проектирования 60-х, 70-х годов.
3. Укажите основные задачи, возлагаемые на автоматизированный способ проектирования.
4. Перечислите основные подсистемы, входящие в структуру САПР.
5. Расскажите о предмете САПР.
6. Перечислите виды обеспечения САПР.
7. Укажите основные принципы построения САПР.
8. Что следует понимать под САПР ЭФУ?
9. Рассмотрите динамику развития САПР ЭФУ и современное состояние этого раздела ускорительной техники.

ГЛАВА 2. ОБЩЕСИСТЕМНЫЕ ВОПРОСЫ САПР, ЦЕЛИ И ЗАДАЧИ ПРОЕКТИРОВАНИЯ

2.1. Краткие сведения о системном анализе

Системный анализ – самостоятельное научное направление, которое интенсивно развивается с середины прошлого столетия. Появление его связано с рядом новых проблем (одной из которых является и проектирование), при решении которых оказались не приемлемы традиционные по тем временам методы.

Уже к концу XIX в. стало резко увеличиваться число комплексных проектов и задач, требующих участия специалистов различных областей знаний. Постоянное усложнение объектов проектирования, развитие науки, появление проблем большой начальной неопределенности, которые невозможно было решить с помощью формальных математических методов прошлого столетия, требовало развития новых подходов при их решении.

Не меньшее влияние на развитие этого направления в науке вызвало возникновение мировых глобальных проблем, связанных с необратимыми катастрофическими последствиями развития цивилизации – экология и разрушение природных балансов, энергетика и захоронение ядерных отходов и т.п.

В этих задачах все большее место стал занимать собственно процесс постановки задачи, возросла роль человека как носителя целостного восприятия, сохранение целостности при расчленении проблемы для облегчения ее решения, роль методов активизации интуиции и опыта специалистов различных областей знаний, принимающих участие при решении сложной проблемы.

Для решения подобных комплексных проблем стало широко использоваться понятие «система», и на определенной стадии развития научных знаний теория систем сформировалась в самостоятельную науку, имеющую непосредственное отношение к САПР. В 30-е годы XX в. возникла теория открытых систем Л. фон Бертана-

лафи. Важный вклад в становление системных представлений внес в начале XX в. А.А. Богданов, предложивший всеобщую организационную науку – *тектологию*. Потребности практики, почти одновременно со становлением теории систем, привели к возникновению направления, названного *исследованием операций*. Это направление возникло в связи с задачами военного характера. Предметом исследований операций является разработка методов анализа целенаправленных действий (операций) и объективная сравнительная оценка решений (поиск оптимальных решений).

В 60-е годы XX в. широкое распространение получил термин «системотехника», предложенный профессором Темниковым из МЭИ, как эквивалент американского *system engineering* (буквальный перевод – системная инженерия), который используется, в основном, в применении системных методов только в технических приложениях (систематология, аналогичный термин, используемый в философской литературе).

В первые годы становления теории систем наибольшее распространение получил термин «*системный подход*», который использовался как методологическое направление философии, так и в прикладном смысле как синоним «комплексный подход».

Наиболее конструктивным из направлений системных исследований в настоящее время считается системный анализ, занимающийся применением методов и моделей теории систем для принятия решений. Впервые понятие системного анализа было предложено в 1948 г. корпорацией RAND (мозговой центр министерства обороны США) в связи с задачами военного управления. Он применяется в тех случаях, когда задача (проблема) не может быть сразу представлена и решена с помощью формальных математических методов. При этом основное внимание уделяется процессу постановки задачи и использованию не только формальных, но и методов качественного анализа. Опирается на основные понятия теории систем и философские концепции, лежащие в основе общесистемных закономерностей.

Рассмотрим некоторые из основных понятий теории систем, которые могут быть нам полезны в дальнейшем: что такое система или сложная техническая система [3–5], о которой уже упоминалось раньше?

Существует несколько десятков определений этого понятия, когда хотят охарактеризовать исследуемый или проектируемый объект, как *нечто целое (единое), сложное, о котором невозможно сразу дать представление, показав его, изобразив практически или описав математическим выражением*. Тем не менее можно определить, что *система* (большая система, система большого масштаба, СТС и т.п.) – *совокупность двух или более элементов, удовлетворяющих следующим условиям:*

- наличие большого количества взаимосвязанных и взаимодействующих между собой элементов, обладающих свойством совместности;
- поведение каждого элемента влияет на поведение целого;
- поведение элементов и их взаимодействие в целом взаимосвязаны;
- сложность функций, выполняемых системой;
- возможность разбиения системы на подсистемы, цели функционирования которых подчинены общей цели;
- наличие управления (часто имеющего иерархическую структуру), разветвленной информационной сети интенсивных потоков информации;
- наличие взаимодействия с окружающей средой (внешней) и функционирование в условиях воздействия внешних факторов.

При формальном подходе к сложной системе *элементом* считается объект, не подлежащий дальнейшему расчленению. Внутренняя структура элемента не подлежит изучению. *Любая совокупность элементов сложной системы может рассматриваться как подсистема*. Обычно подсистемой является некоторая самостоятельно функционирующая часть системы. Правильное выделение подсистемы в СТС часто способствует упрощению расчетов при исследовании и обеспечивает большую наглядность при интерпретации результатов.

Основное свойство системы – самоорганизация (лобализация) или, можно сказать, устойчивость. В широком смысле под этими словами понимается *способность системы возвращаться в состояние равновесия после того, как она была из этого состояния выведена под влиянием внешних сил, или способность после воздействия внешней среды путем последовательного изменения своих*

свойств прийти к некоторому устойчивому состоянию (предполагается, что воздействие оказывается в некоторых допустимых пределах).

Осознание всех вышеперечисленных проблем позволило американскому математику Р. Акроффу назвать их основным источником интеллектуальной революции, вызывающей смену эпох. Уходящую эпоху он назвал веком машин, а наступающую веком систем. Появление этого направления в науке привело к развитию специфических направлений в математике – теории массового обслуживания, теории игр, исследования операций, математического программирования, дискретной математики, значительно расширилась сфера применения теории вероятностей и математической статистики.

2.2. Блочно-иерархический подход – основной метод исследования сложных систем

Блочно-иерархический подход не является каким-то неожиданным изобретением. Он явился естественной реакцией человечества на сложность в исследовании и описании различных больших по объему, многокомпонентных с большим количеством связей объектов. Его фрагменты в виде блок-схем алгоритмов, структурных схем различных технических объектов и т.п. традиционно использовались для облегчения понимания и разработки в инженерной практике.

Дело в том, что описание технических объектов должно быть согласовано (по сложности) с возможностями восприятия человека. Однако выполнить это требование в рамках одного описания, не расчленяя его на некоторые составные части, удастся лишь для простых изделий. В СТС, как правило, требуется структурирование описаний и соответствующее расчленение представлений о проектируемом объекте на иерархические уровни и аспекты. Это позволяет распределить работы проектирования между подразделениями проектной организации, что способствует повышению эффективности и труда проектировщиков.

Разделение описаний по степени детализации отображаемых свойств и характеристик объектов лежит в основе блочно-иерархического подхода к проектированию и приводит к появле-

нию иерархических уровней (уровней абстрагирования) в представлении о проектируемом объекте.

На каждом уровне используются свои понятия систем и элементов. При этом элементы верхнего уровня иерархии являются системами для нижнего уровня. Подобное разбиение продолжается вплоть до получения на некотором уровне элементов, описание которых дальнейшему расчленению не подлежит. Такие элементы называются *базовыми*. Таким образом, в основе блочно-иерархического подхода используются два принципа:

- принцип иерархичности, означающий структурирование представлений об объектах проектирования по степени детализации описаний;
- принцип декомпозиции (блочности) – разбиение представлений каждого уровня на ряд составных частей (блоков) с возможностями раздельного (не блочного) проектирования объектов.

Кроме расчленения описаний по степени подробности, для отображения свойств проектируемого объекта, порождающих иерархические уровни (которые иногда называют *горизонтальными*), используется декомпозиция описаний объекта по характеру отображаемых свойств, что приводит к появлению ряда так называемых аспектов описаний.

В соответствии с ЕСКД вводятся функциональный, конструкторский и технологический аспекты, которые считаются вертикальными уровнями. Решения задач проектирования на этих уровнях называют, соответственно, функциональным, конструкторским и технологическим проектированием.

Функциональный аспект связан с отображением основных принципов функционирования, характера физических и информационных процессов, протекающих в объекте, и находит отображение в принципиальных электрических, кинематических, пневматических и т.п. схемах и сопровождающих их документах.

Конструкторский аспект связан с реализацией результатов функционального аспекта, т.е. определением геометрических форм и их взаимным расположением в пространстве (принципиальная электрическая (логическая) схема – функциональный аспект; печатная плата, реализующая эту схему, – конструкторский аспект).

Технологический аспект связан с описанием методов и средств изготовления конструкции.

Все перечисленные аспекты тесно взаимосвязаны. И хотя основные идеи разработки определяются функциональным аспектом, удачное конструктивное решение в значительной степени определяет перспективы внедрения разработки. Еще большее значение, особенно на современном этапе, имеет технологический аспект. Использование новейших технологических достижений может значительно повлиять как на функциональный, так и на конструкторский аспекты.

Внутри каждого аспекта, возможно, свое специфическое выделение иерархических уровней.

В рамках рассмотренной блочно-иерархической структуры возможны два подхода к проектированию, которые и реализуются на практике. Если решение задачи высоких иерархических уровней предшествует решению задач более низких уровней, то проектирование называется *нисходящим*. Если раньше выполняются этапы, связанные с нижними иерархическими уровнями (т.е. проектирование идет от готовых базовых элементов, не подлежащих расчленению и разработке), проектирование называют *восходящим*.

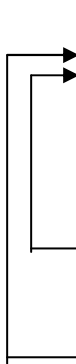
При всех преимуществах блочно-иерархического подхода к проектированию, позволяющему решать задачи большой размерности, основным недостатком как нисходящего, так и восходящего проектирования является то, что на каждом этапе работа ведется с неопределенными до конца объектами, а решения принимаются в обстановке неполной информации. При проектировании сверху вниз может оказаться, что базовых элементов на самом нижнем уровне не существует в природе, а их разработка весьма проблематична. Проектирование от базовых элементов (снизу вверх) может привести к тому, что, подойдя к самому верхнему уровню, оказывается, что невозможно удовлетворить техническому заданию на разрабатываемое изделие.

Тем не менее более разумной альтернативы блочно-иерархическому подходу не существует, поэтому он является основным методом разработки сложных технических систем.

2.3. Жизненный цикл сложной технической системы (ГОСТ 22487–77)

Рассмотрим соединенные части процедуры проектирования и как они изменяются со временем. Основные этапы проектирования

и задачи, решаемые на этих этапах, принято называть *маршрутами проектирования*, определяющими жизненный цикл сложной технической системы, который включает следующую последовательность этапов и проектных процедур.

- 
1. Предпроектные исследования.
 2. Техническое задание.
 3. Техническое предложение.
 4. Эскизный проект.
 5. Технический проект.
 6. Рабочий проект.
 7. Изготовление опытного образца.
 8. Опытная эксплуатация.
 9. Изготовление серийного образца.
 10. Промышленная эксплуатация.
 11. Модернизация.

Предпроектные исследования выполняются заказчиком от момента появления идеи до того, как будут сформулированы конкретные задачи проектирования. Это этап научно-исследовательской работы (НИР) – поиск принципиальных возможностей построения системы, исследование новых принципов, технических средств, обоснование общих решений.

Техническое задание также формулирует заказчик. Устанавливаются основное назначение изделия, тактико-технические характеристики, показатели качества и технико-экономические требования, предъявляемые к изделию, определение этапов разработки. При отсутствии стандартов на изделие разработки технических условий (ТУ) для серийного и массового производства определяются технические требования, правила приемки, методы контроля (испытаний), транспортировки и хранения, указания по эксплуатации, гарантии поставки и т.п.

Техническое предложение – предэскизное проектирование – выполняет разработчик, оно содержит:

- технико-экономическое обоснование, определение стоимости разработки;
- выяснение вариантов возможных решений;
- проверку на патентную чистоту и конкурентоспособность (если аналогов нет, то обязательна заявка на изобретение, если не конкурентоспособна – не следует разрабатывать).

Эскизный проект подлежит разработке, если это предусмотрено техническим заданием. Разрабатывается с целью установления принципиальных (конструкторских, схемных и др.) решений, дающих общее представление о принципах работы изделия, и включает следующие работы:

- выполнение вариантов возможных решений (их конструктивные разработки);
- предварительное решение вопросов установки и транспортировки изделий;
- изготовление и испытание макетов с целью проверки принципов работы (натурные испытания) и т.д.

Технический проект включает совокупность конструкторских документов, содержащих окончательные технические решения, дающие полное представление об устройстве изделия и исходных данных для разработки технической документации.

Технический проект может предусматривать разработку вариантов отдельных частей изделия. В этом случае выбор оптимального варианта производится на основе испытания опытных образцов изделия. Разрабатывается он, если это предусмотрено техническим заданием, и состоит из:

- разработки конструкторских решений изделия и его составных частей;
- выполнения необходимых расчетов;
- выполнения необходимых схем (принципиальных схем, схем соединения и т.п.);
- анализа конструкции на технологичность;
- разработки, изготовления и испытания макетов.

Рабочий проект (рабочая документация) – совокупность конструкторских документов, необходимых для изготовления и испытания опытного образца. Объем и содержание рабочей документации определяются видом изделия (деталь, сборная единица, комплекс и т.д.) и масштабом производства – опытное, серийное, массовое.

Этапы 7–11 – этапы жизненного цикла СТС, хотя и не являются непосредственными этапами проектирования, тем не менее тесно с ним связаны. Прохождение этих этапов, как правило, приводит к очередной итерации процедуры проектирования или отдельных ее частей.

Так, недостатки, выявленные на этапе опытной эксплуатации (этап 8), что вполне естественно, приводят к необходимости вносить изменение в проект, проводя дополнительные проектные работы.

Промышленная эксплуатация также имеет временные ограничения по причинам, рассмотренным в гл. 1. В связи с этим наступает новый этап проектирования, связанный с модернизацией изделия.

2.4. Формализация процесса проектирования

Введение автоматизации в процедуру проектирования на основе средств вычислительной техники, т.е. кибернетизация проектирования, требует точного формального алгоритма этой процедуры. Очень часто процедура проектирования представляется как изготовление чертежно-технической документации, что, на самом деле, является лишь завершающим этапом этого процесса – вершиной «айсберга». Этому этапу предшествует решение ряда гораздо более сложных задач, которые приходится выполнять проектировщикам.

Независимо от способа проектирования (автоматизированное или неавтоматизированное) и предметной области объектов проектирования процедура проектирования включает ряд традиционных (обязательных) этапов. На рис. 2.1 в рамках рассмотренного ранее блочно-иерархического подхода представлена блок-схема проектирования на отдельном уровне иерархии.

После получения технического задания от предшествующего уровня иерархии наступает этап проектирования, который называется *синтезом технических объектов*. Он включает в себя выбор структуры объекта – структурный синтез, создание модели объекта и выбор его параметров (параметрический синтез).

Если среди вариантов структуры изделия ищется не любой приемлемый вариант, а наилучший (в некотором смысле), задачу синтеза называют *структурной оптимизацией*. Выбор параметров структуры, оптимальных с позиции некоторого критерия при заданной структуре объекта, называют *параметрической оптимизацией*.

Следующий этап проектирования – анализ технических объектов. На этом этапе происходит исследование свойств выбранной структуры объекта проектирования (его модели) при различных внутренних параметрах.

Синтез и анализ находятся в диалектическом единстве и не могут существовать друг без друга.

Результаты анализа определяют завершающий этап проектирования – изготовление технической документации.

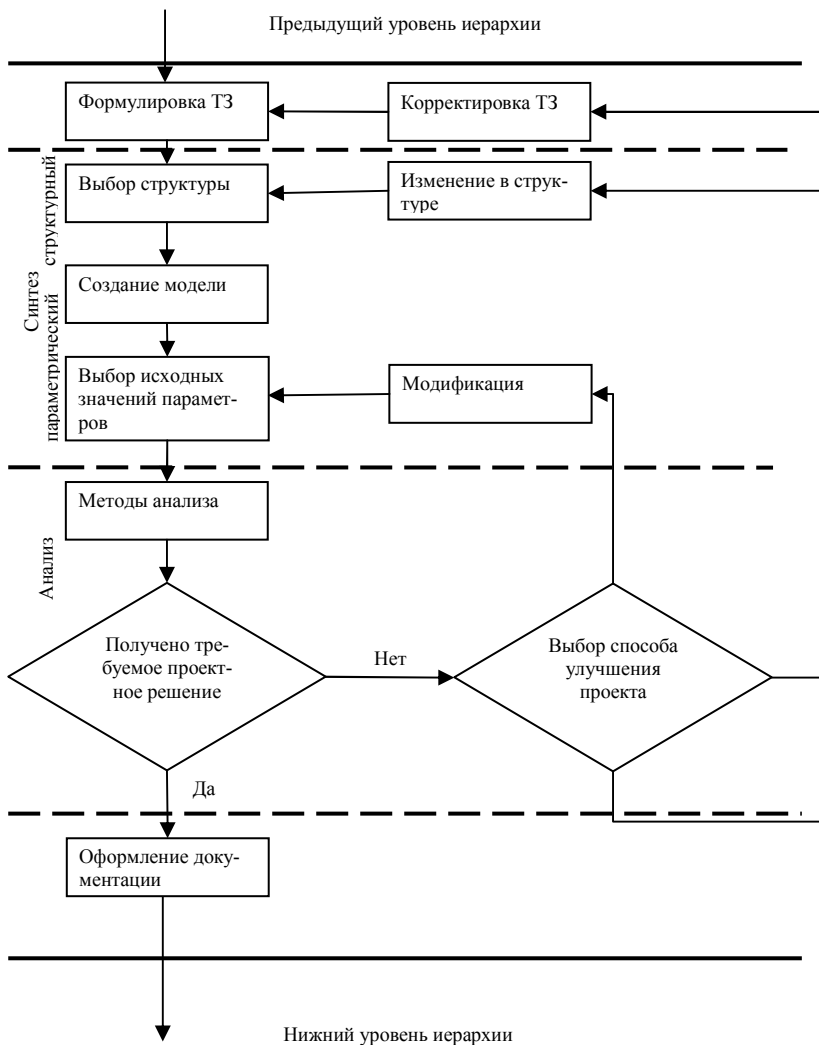


Рис. 2.1. Кибернетическая схема проектирования

Если проектное задание не достигнуто, то выбирается способ его достижения. Наиболее простой вариант улучшения проекта – изменение внутренних параметров выбранной структуры объекта (параметрический синтез). Если эти изменения не приводят к же-

лаемому результату, необходимо изменять структуру объекта (структурный синтез). Когда же и этот способ не приводит к выполнению ТЗ, необходимо обращаться на верхний уровень иерархии проекта с целью облегчения ТЗ.

Завершающий этап проектирования – изготовление технической документации.

Таким образом, процедура проектирования включает три основных этапа: синтез технических объектов, их анализ и изготовление технической документации. Причем первые две задачи находятся в тесном взаимодействии, и в рамках автоматизированного проектирования их основу составляет математическое моделирование.

Синтез технических объектов нацелен на создание новых вариантов объекта проектирования и представляет наиболее сложную математическую задачу, основанную на разделах математики, ориентированных на вычислительные методы, требующих очень больших ресурсов вычислительных систем. Наиболее проблемным является структурный синтез. Основу параметрического синтеза составляют методы математического программирования (методы оптимизации), которые интенсивно развиваются на протяжении последних десятилетий.

Анализ технических объектов – это изучение их свойств. Результаты анализа должны ответить на вопросы: какими свойствами обладает исследуемый объект; насколько хорошо он удовлетворяет предъявляемым требованиям, но непосредственно не содержит в себе рецептов относительно того, что нужно сделать, чтобы улучшить объект и выполнить проектное задание? Но, несмотря на такой пассивный характер результатов анализа, роль задачи анализа велика, поскольку результаты этого анализа составляют основу реализации задач синтеза. Задачи синтеза связаны с выбором оптимальной стратегии изменения параметров (или структуры) объекта с целью получения результатов, удовлетворяющих ТЗ (в простейшем случае – переборные алгоритмы). Оценка результатов синтеза проводится с помощью методов анализа, т.е. синтез осуществляется через анализ и невозможен без него. Обычно различают два основных метода анализа технических объектов.

Одновариантный анализ (или *детерминированный*), который включает следующие разновидности:

- анализ статических состояний (без учета изменений параметров во времени);
- анализ переходных процессов;
- анализ стационарных видов колебаний;
- анализ частотных характеристик;
- анализ устойчивости.

Многовариантный анализ заключается в исследовании объекта в некоторой окрестности отображаемой точки, он связан с изучением влияний случайных факторов и внешней среды на работу объекта проектирования и включает, как правило, две задачи:

- анализ чувствительности выходных параметров к изменениям внутренних и изменениям параметров внешней среды;
- статистический анализ, основу которого составляют методы теории вероятностей и математической статистики (более подробно методы анализа и синтеза будут рассмотрены в дальнейшем).

Контрольные вопросы

1. Каковы причины появления и основные направления внедрения методов системного анализа?
2. Что понимается под словом «системы»?
3. Укажите основной метод исследования сложных систем и опишите его структуру.
4. Перечислите основные аспекты проектирования и кратко их охарактеризуйте.
5. Укажите основные недостатки блочно-иерархического подхода при проектировании.
6. Опишите основные составляющие жизненного цикла сложной технической системы. Какие три основные задачи решаются при проектировании?
7. Что такое синтез при проектировании и каковы его разновидности?
8. Что есть анализ при проектировании и каковы его разновидности?
9. Каковы особенности многовариантного анализа?

ГЛАВА 3. МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САПР (МЕТОДЫ МНОВОВАРИАНТНОГО АНАЛИЗА)

Введение

Вопросы математического обеспечения представляют центральную и наиболее сложную часть автоматизации проектирования. Необходимость современных подходов к проектированию потребовала как расширения сферы использования и развития традиционных разделов математики, так и разработки новых, ориентированных на реализацию с помощью средств современной вычислительной техники. Все основные этапы в САПР – анализ, синтез и изготовление технической документации связаны с использованием современных математических методов.

Математическое обеспечение САПР имеет иерархическую структуру и в определенной степени соответствует иерархии задач проектирования сложных технических систем.

Можно выделить два уровня математического обеспечения САПР, по аналогии с двумя уровнями (универсальным и специализированным) различных видов обеспечения САПР. Нижний уровень средств математического обеспечения отражает специфику предметной области и используется при анализе разрабатываемых технических систем. Так, при расчете электронных схем основу математического обеспечения определяют законы теории электрических цепей – это законы Ома, Кирхгофа, а также теория графов и ряд традиционных разделов математики, дифференциальные уравнения, линейная алгебра и т.п. Если говорить о САПР ЭФУ, то здесь основным аппаратом уравнения Максвелла, законы релятивистской механики, разнообразные дифференциальные и интегральные уравнения и т.п. Естественно, что каждый из перечисленных фундаментальных законов физики и математических методов их реализации имеет свои специфические особенности для различных предметных областей, которые и отражаются в особенностях мате-

математического обеспечения САПР этих областей. Именно этим особенностям и будет посвящена вторая часть этого пособия – методическое обеспечение САПР ЭФУ.

Однако наряду с вышеуказанным, имеется ряд математических методов, которые используются на верхних уровнях иерархии и имеют достаточно универсальный характер. К таким методам, прежде всего, относятся методы, применяемые для решения задач синтеза – различные разделы математического программирования, вычислительная геометрия и геометрическое моделирование и т.п.

При анализе технических объектов достаточно универсальными являются методы многовариантного анализа – статистический анализ (метод Монте-Карло). Такой вид анализа определяется как имитационное моделирование.

Эти методы и будут рассмотрены в настоящей главе, но вначале следует определиться с понятиями модели и моделирования, основополагающими понятиями САПР.

3.1. О моделях и моделировании

Под *моделью* принято понимать такой материальный или абстрактный объект, который в процессе изучения заменяет объект-оригинал, сохраняя важнейшие типичные его черты.

Что касается моделирования, то современные словари дают следующее определение этому термину [5]: «*Моделирование* – исследование каких-либо явлений, процессов или систем путем построения и изучения их моделей, использование этих моделей для определения или уточнения характеристик или рационализации способов вновь конструируемых объектов». А если совсем коротко, то построение модели и исследование объектов на их моделях называется *моделированием*.

С незапамятных времен при изучении сложных процессов, явлений, конструирования новых сооружений человек применяет модели. Хорошо построенная модель, как правило, доступнее для исследования, нежели реальный объект.

Существует несколько приемов моделирования, которые можно условно объединить в две большие подгруппы: материальное (предметное) и идеальное (абстрактное) моделирование.

К материальному относятся такие способы моделирования, при которых исследование ведется на основе модели, воспроизводящей основные геометрические, физические, динамические и функциональные характеристики изучаемого объекта.

Основными разновидностями материального моделирования является физическое и аналоговое моделирование.

Физическим принято называть моделирование, при котором реальному объекту противопоставляется его увеличенная или уменьшенная копия, допускающая исследование (как правило, в лабораторных условиях) с помощью последовательного перенесения свойств изучаемых процессов и явлений на объект на основе использования теории размерности физических величин и теории подобия.

Физическое моделирование широко используется в науке и технике как способ разработки и экспериментального изучения на моделях разнообразных объектов, самолетов (продувка в аэродинамической трубе), судов, тепловых установок и т.п. Этот вид моделирования применяется и при разработке установок в электрофизике. Так, при проектировании 20-километрового протонного синхротрона в Протвино в Радиотехническом институте была создана его уменьшенная (100-метровая) копия, на которой изучались элементы конструкции разрабатываемой установки.

Преимущества физического моделирования перед натурным экспериментом заключаются в том, что условия реализации процесса модели могут значительно отличаться от условий, свойственных оригиналу, и их выбирают из удобства и простоты исследования.

Аналоговое моделирование основано на аналогии процессов и явлений, имеющих различную физическую природу, но одинаково описываемых формально (одними и теми же математическими уравнениями). Наиболее простой пример – изучение механических колебаний с помощью электрической схемы, описываемой теми же дифференциальными уравнениями. Электрическое моделирование лежит в основе работы машин непрерывного действия – так называемых аналоговых или моделирующих машин (например, дифференциального анализатора, электрического интегратора и др.). Моделью какого-либо процесса при таком «машинном моделировании» можно, очевидно, с равным основанием называть как про-

грамму этого процесса, так и саму машину. В 60-х годах прошлого века на кафедре ЭФУ МИФИ для исследования динамики частиц в линейных ускорителях использовалась аналоговая машина «Гамма». Однако, справедливости ради, следует отметить, что этот вид моделирования для подобных задач в современных условиях полностью вытеснен цифровым компьютерным моделированием, обладающим гораздо большей точностью и достаточно высоким быстродействием (основного достоинства аналоговых машин). По этим же причинам утратили актуальность и так называемые гибридные вычислительные системы – объединение аналоговых и цифровых машин.

Необходимо отметить, что в обоих типах материального моделирования (физического и аналогового) модели являлись материальным отражением исходного объекта и были связаны с ними своими геометрическими, физическими и другими характеристиками, причем процесс исследования был тесно связан с материальным воздействием на модель, т.е. состоял в натурном эксперименте с ней. Таким образом, предметное моделирование по своей природе является экспериментальным.

От предметного моделирования принципиально отличается идеальное (абстрактное), которое основано не на материальной, а на идеальной, мыслимой аналогии. Идеальное моделирование носит теоретический характер. Различают два вида идеального моделирования: интуитивное и знаковое.

Под *интуитивным* понимается моделирование, основанное на интуитивном представлении об объекте исследования, не поддающимся формализации, или не нуждающимся в ней. В этом смысле жизненный опыт каждого человека может считаться его индивидуальной моделью окружающего мира. Этот вид моделирования особенно развит у музыкантов, ученых, шахматистов.

Знаковым называют моделирование, использующее в качестве моделей знаковые преобразования какого-либо вида: схемы, графики, чертежи, формулы, наборы символов и т.п., а также включающее набор знаков, по которым можно оперировать с wybranymi знаковыми образованиями и их элементами.

Важнейшим видом знакового моделирования является математическое моделирование, при котором исследование объекта осуществляется посредством модели, сформулированной на языке ма-

тематики, и использованием тех или иных математических методов. Классическим примером математического моделирования является описание и исследование законов механики И. Ньютона средствами математики.

3.2. Математическое моделирование

Математическое моделирование есть приближенное описание какого-либо класса явлений внешнего мира с помощью математических моделей, под которыми понимается совокупность математических объектов (чисел, символов, переменных, векторов, множеств и т.п.) и отношений между ними, которые адекватно отображают основные свойства проектируемых объектов.

В общем виде математическая модель может быть представлена в виде зависимости (далеко не всегда функциональной):

$$Y = F(X, Q),$$

где Y – вектор выходных параметров; X – вектор входных (внутренних) параметров (параметров элементов); Q – вектор внешних параметров.

Даже если существует функция $F(X, Q)$, это еще не означает, что она известна проектировщику. В большинстве случаев связь между внутренними (входными) и внешними (выходными) параметрами задается в алгоритмической форме, например через численное решение системы уравнений для некоторых параметров, которые характеризуют состояние объекта и не относятся к перечисленным выше параметрам (выходным, внутренним и внешним). Они называются *фазовыми переменными* (как бы промежуточные параметры). Вектор фазовых переменных задает точку в пространстве, называемом *фазовым*. Фазовые переменные, непосредственно характеризующие запасы энергии в элементах системы, именуют переменными состояния. Выходные параметры при таком подходе имеют смысл функционалов, зависимостей фазовых переменных от времени (фазовые переменные возникают только при анализе моделей).

В зависимости от характера отображаемых свойств проектируемого объекта модели можно разделить на структурные (связанные, прежде всего, с геометрическим моделированием) и функциональные. Структурные модели отображают только структурные (в ча-

ственном случае – геометрические) свойства объекта. Структурные модели обычно используются в случаях, когда задачи структурного анализа удастся ставить и решать, абстрагируясь от особенностей физического процесса в объекте.

Наибольшую группу представляют так называемые функциональные модели. Они, в основном, ориентированы на методы анализа. Традиционными и давно разрабатываемыми являются теоретические модели. Их получают на основе изучения физических закономерностей, а структура уравнений и параметров этих моделей определяются их физическим толкованием. Они основаны, как правило, на фундаментальных физических законах. Так, методы математической физики включают те уравнения, которые применяются для построения и изучения математических моделей, описывающих большие классы физических явлений. Теоретические модели по своей структуре разделяют на *линейные* и *нелинейные* (в зависимости от линейности или нелинейности входящих в них уравнений), *непрерывные* и *дискретные* (в зависимости от непрерывности или конечной счетности их решений), *динамические* и *статические* (имеется или нет зависимость от времени).

К сожалению, не всегда условия применимости моделей и методов могут быть найдены и исследованы строгими методами.

Часто применимость того или иного компонента математического обеспечения зависит от конкретных условий, многообразие которых не поддается исчерпывающему учету и классификации.

Особый класс математических *функциональных* моделей составляют так называемые *имитационные*, которые иногда называются *формальными* или даже *эвристическими*. Развитие этого направления моделирования связано с разработкой сложных технических систем и ограничениями традиционных теоретических моделей. Эти модели получают на основе проявления свойств моделируемого объекта во внешней среде, модель типа «черный ящик».

В соответствии с иерархической структурой сложных технических систем (элемент, подсистема, система) математические модели разделяются на три уровня: низший (микро), средний (макро) и высший [6].

На низшем уровне, называемом также уровнем проектирования базовых элементов, (или микроуровне) математические модели должны отражать процессы, протекающие в общем случае в трех-

мерной сплошной среде. Они представляют собой дифференциальные уравнения в частных производных и часто называются *распределенными* моделями. Они основаны на известных фундаментальных физических законах (уравнения Ламе для механики упругих сред, Навье – Стокса для гидравлики, уравнения теплопроводности – термодинамики, уравнения Максвелла – электромагнитные поля). В качестве фазовых переменных в этих уравнениях выступают напряженности полей и плотности потоков. Точное решение этих уравнений (аналитическое) удастся получить лишь для ограниченного числа частных случаев, поэтому основная задача, возникающая при моделировании, состоит в построении приближенной дискретной модели. Для этого используются методы конечных разностей, конечных элементов или интегральных граничных элементов. Так как получаемая при дискретизации пространства аппроксимирующая система алгебраических уравнений имеет очень высокий порядок, то при моделировании достаточно сложных технических объектов приходится принимать ряд допущений и переходить к моделированию на среднем (макро) уровне.

Элементами этого уровня являются объекты, которые на микроуровне рассматриваются как системы. Математические модели на данном уровне представляют обыкновенные дифференциальные уравнения, в частных случаях статических задач они превращаются в алгебраические и трансцендентные уравнения. В качестве фазовых переменных выступают токи и напряжения в электрической системе (вместо плотности тока и напряженности полей), в механической системе аналогами этих переменных являются силы и скорости, в гидравлической системе – массовые расходы и давление, а в тепловых – тепловые потоки и температура.

Математическую модель системы на этом уровне получают объединением компонентных и топологических уравнений.

Законы функционирования элемента подсистемы задаются компонентными уравнениями, связывающими, как правило, разнообразные фазовые переменные, относящиеся к данному элементу.

Компонентные уравнения могут быть линейными или нелинейными, алгебраическими, обыкновенными дифференциальными уравнениями или интегральными.

Связь между однородными фазовыми переменными, относящимися к различным элементам подсистемы, задается топологиче-

скими уравнениями, получаемыми на основании сведений о структуре системы.

В большинстве технических систем можно выделить три типа простейших элементов (компонент):

элемент типа R – элемент диссипации энергии (на этом элементе, как правило, происходит преобразование энергии в тепловую (электрические сопротивления), трения в механических системах, вязкость в жидкостях и газах и т.п.);

элементы C и L , в которых происходит накопление потенциальной и кинетической энергии.

Для каждой физической подсистемы характерны свои законы, однако для простейших элементов форма выражающих их уравнений оказывается одинаковой.

Например, для электрической подсистемы фазовыми переменными являются токи I и напряжения U , а компонентные уравнения для простейших элементов выглядят следующим образом:

для сопротивления (законы Ома) $I = U / R$;

для емкости $I = C(dU / dt)$, где C – электрическая емкость;

для индуктивности $U = L(dI / dt)$, где L – электрическая индуктивность.

Топологические уравнения в большинстве случаев базируются на уравнениях равновесия и уравнениях непрерывности.

Так, для электрической подсистемы связь между отдельными элементами этой подсистемы устанавливается на основании законов Кирхгофа.

Уравнения первого закона Кирхгофа устанавливают равенство нулю суммы токов в узлах системы, т.е. $\sum_{k \in p} I_k = 0$ (уравнения равновесия), где I_k – ток в k -й ветви; p – множество номеров ветвей, инцидентных рассматриваемому узлу.

Из уравнения второго закона Кирхгофа видно, что сумма падений напряжений на элементах схемы при их обходе по произвольному контуру равна нулю, т.е. $\sum_{j \in q} U_j = 0$ (уравнение непрерывности), где j – номер ветви; U_j – падение напряжения на j -й ветви схемы, входящей в контур; q – множество номеров ветвей, входящих в рассматриваемый контур.

Аналогичные компонентные и топологические уравнения можно привести для механических, гидравлических (пневматических) и тепловых систем.

Модели высшего уровня или модели технических систем, которые также называют информационными, применяются для сложных устройств и комплексов (вычислительные комплексы, радиолокационные станции, ускорительные комплексы и т.п.), функционирование которых рассматривается как цепь событий, происходящих в дискретные моменты времени и заключающихся в изменении состояний элементов.

Для построения математических моделей высшего уровня широко используют математическую логику, теорию массового обслуживания, методы теории автоматического управления, теорию графов и т.п. Очень широкое применение на этом уровне имеет упоминавшееся ранее имитационное моделирование, что представляет собой дословный перевод английского выражения «Simulation modeling». Как легко убедиться, перевод сделан не слишком хорошо, так как в нем содержится тавтология (любое моделирование — есть имитация). Однако этот термин в отечественной практике так широко распространился, что хоть он и неудачен, но маловероятно, что он претерпит изменение.

Суть метода имитационного моделирования состоит в том, что процесс функционирования сложной системы представляется в виде определенного алгоритма, который и реализуется на ЭВМ. По результатам реализации могут быть сделаны те или иные выводы относительно исходного процесса.

Наряду с широко используемыми, традиционными методами математического моделирования (теоретическими), большая роль при проектировании отводится и этому виду, причем его появление, как указывалось ранее, связано с возрастающими сложностями объектов, процессов, явлений, ввиду отсутствия в настоящее время математического аппарата, пригодного для исследования. Кроме того, реальные системы подвержены влиянию различных случайных факторов. Учет этих факторов аналитическим путем представляет весьма большие трудности, зачастую непреодолимые при большом их числе. Этот вид моделирования реализуется на компьютере и отображает внутренние и внешние взаимосвязи моделируемого сложного объекта.

Имитационное моделирование можно трактовать как результат объединения двух подходов: дедуктивного и индуктивного. Дедуктивный подход или подход «сверху вниз» предполагает определение структуры модели на основе знания физических законов (традиционные теоретические модели), которым подчиняются происходящие в системе процессы и построение системы с последующим уточнением параметров. Индуктивный же подход, или подход «снизу вверх», оперирует с экспериментальными данными типа «вход-выход», не используя знания внутренних свойств системы. Типичным примером такого подхода служит экспериментирование с моделью на основе метода Монте-Карло (который рассматривается далее). При этом имитационное моделирование представляет собой численный эксперимент сродни физическому эксперименту. В ходе эксперимента варьируются экзогенные (внешние) параметры модели, результаты представляются в статистическом смысле, что неизбежно при работе со сложной системой. Основные свойства системы и состояние ее параметров определяются на основе длительного наблюдения и набора соответствующих статистических закономерностей.

Поскольку статистическое моделирование является одним из основных приемов имитационного моделирования, носит универсальный характер (его идея очень проста и независима от приложений), широко применяется в техники и физики, то далее будет более подробно рассмотрена его методика.

Среди общих требований, которые не зависят от типа моделей и вида моделирования, следует отметить следующие.

Точность математической модели – свойство, определяющее степень совпадения предсказанных с помощью модели выходных параметров объекта с истинными значениями этих параметров. Количественная оценка точности затруднена по следующим причинам:

реальная модель, как правило, обладает множеством выходных параметров (поэтому возникает необходимость сведения вектора параметров к скалярному представлению с помощью не всегда достаточно обоснованных методов);

модель составляется, как правило, для многократного использования, при анализе различных вариантов объекта (однако характер проявления тех или иных свойств объекта зависит от особенностей взаимодействия с внешней средой и другими объектами);

истинное значение (которое предполагается заранее известным) обычно отождествляется с экспериментом, однако погрешность экспериментов может оказаться соизмеримой с погрешностями модели, а часто и превышает их.

Экономичность – отождествляется, прежде всего, затратами машинного времени при реализации на компьютере. Несмотря на постоянный рост производительности современных вычислительных систем, потребности задач (в том числе и проектных) также возрастают, зачастую превышая возможности машин, в связи с этим разработчикам приходится заботиться об экономичности алгоритмов.

Степень универсальности модели связана с тем, что разработка модели и ее адаптация к вычислительной среде – достаточно длительный и дорогостоящий процесс, поэтому расширение сферы приложения модели повышает эффективность разработки и снижает стоимость разработанного программного продукта.

Требования точности, универсальности и экономичности модели достаточно противоречивы и требуют принятия оптимальных решений в каждом конкретном случае.

Кроме перечисленных, существует ряд специфических требований к математическим моделям различных типов. Так, требования к традиционным теоретическим моделям определены в курсах вычислительной математики (это корректность, устойчивость, сходимости, обусловленность и т.п.), где с ними можно подробно ознакомиться.

3.3. О методах многовариантного анализа

Одновариантный анализ служит для оценки выходных параметров при заданных внутренних и внешних параметрах. Он является необходимой составной частью более сложных задач многовариантного анализа, оптимизации и структурного анализа. В этих случаях задачи одновариантного анализа приходится решать многократно, что значительно усложняет процедуру проектирования.

Особенность методов многовариантного анализа (так же как и методов оптимизации) заключается в отсутствии их «жесткой привязки» к конкретным приложениям. Они являются, в определенном смысле, некоторой надстройкой над методами одновариантного анализа.

Важной задачей методов многовариантного анализа при проектировании является учет влияния малых случайных отклонений внутренних параметров объекта и внешней среды на его работу, т.е. определение значений допустимых отклонений этих параметров от их номинальных значений (определение допусков).

Очень часто такого вида отклонения определяют надежность работы установки и точность поддержания ее выходных характеристик. Проведение многовариантного анализа предполагает изучение случайных событий и процессов, происходящих в различных системах проектируемых изделий, использование аппарата теории случайных процессов и математической статистики для оценки их влияния.

При проектировании электрофизических установок и ускорителей указанные отклонения параметров могут значительно понизить эффективность использования пучка заряженных частиц и даже привести к его потере.

Исследование влияния малых отклонений внутренних (и внешних параметров) на выходные характеристики разрабатываемых изделий предполагает исследование характера (типа) отклонений этих параметров и знание статистических характеристик случайных отклонений.

По типу случайные отклонения делятся на пространственные (постоянные во времени, например отклонения в параметрах элементов электронных схем сопротивлений, конденсаторов, неточности в изготовлении ячеек ускоряющих волноводов и т.п.) и временные – изменяемые во времени.

Временные отклонения, в свою очередь, могут разделяться на медленные, импульсные (постоянные в пределах одного импульса), внутриимпульсные и т.п.

На практике применяются следующие методы учета такого вида отклонений:

- метод наихудшего случая;
- метод граничных испытаний;
- метод моментов;
- метод натурных испытаний;
- метод статистических испытаний (метод Монте-Карло).

Далее будет дана краткая характеристика первых четырех методов. Метод Монте-Карло, который получил очень широкое распро-

странение в физических исследованиях и техники (и не только в аспекте допусков), будет рассмотрен более подробно.

1. *Метод наилучшего случая.* Он характеризуется следующими требованиями: выходной параметр должен находиться в пределах установленного поля допусков при наиболее неблагоприятных сочетаниях погрешностей. Существует два способа расчета погрешностей этим методом.

Первый состоит в том, что погрешность внутренних параметров определяется арифметическим или квадратичным суммированием частичных отклонений, вызванных действием каждого дестабилизирующего фактора в отдельности. По этим суммарным погрешностям определяется погрешность выходного параметра.

При втором способе отдельно определяется частичная погрешность выходного параметра за счет погрешностей внутренних, вызванных влиянием каждого воздействующего фактора в отдельности. Результирующая погрешность выходного параметра определяется суммированием частичных погрешностей.

В первом случае учет этих изменений может быть произведен следующим образом:

$$\Delta Q_{\Sigma} = \Delta Q_1 + \Delta Q_2 + \Delta Q_3 + \Delta Q_4$$

или

$$\Delta Q_{\Sigma} = \Delta(\Delta Q_1^2 + \Delta Q_2^2 + \Delta Q_3^2 + \Delta Q_4^2)^{1/2}.$$

При этом изменения (отклонения) энергии оцениваются как

$$\Delta W = f(\Delta Q_{\Sigma}), \quad \Delta W = \sum_{i=1}^4 f(\Delta Q_i).$$

Следует отметить, что оба способа дают одинаковые результаты только при линейной зависимости параметров. Для нелинейных зависимостей при втором способе получаются меньшие погрешности выходного параметра.

Основные недостатки этого метода следующие:

необосновано арифметическое и квадратичное суммирование погрешностей параметров (квадратичное суммирование частичных погрешностей внутренних параметров справедливо только при нормальном законе распределения погрешностей);

отсутствует количественная оценка попадания выходного параметра в поле допусков;

невозможно оценить случаи появления крайних и средних значений выходного параметра;

не позволяет определить причину выхода параметра из поля допуска, если на него влияют несколько факторов.

Однако этот метод очень прост и позволяет быстро оценить (хотя и грубо) верхний предел допустимых отклонений внутренних и внешних параметров изделия.

2. *Метод граничных испытаний.* В этом методе обычно фиксируются параметры ряда элементов. Параметры других элементов варьируются с учетом требований к фиксируемым параметрам.

Из двух варьируемых параметров один, имеющий наибольшее влияние (степень влияния характеризуется частной производной), является основным.

Делался ряд расчетов, при которых фиксируется ряд значений одного из параметров, и в широких пределах проводится изменение основного параметра. С помощью этих расчетов определяются допустимые (граничные) значения изменений этих параметров.

По результатам расчетов строятся графики. По одной из осей всегда откладываются значения основного параметра, по другой – второй изменяемый параметр. Графики представляют область устойчивости работы, внутри которой значения выходного параметра соответствуют требуемым.

По графикам выбирают номинальные значения и допустимые отклонения основного и варьируемого параметров так, чтобы величины выходного параметра находились внутри области устойчивости.

Этот метод используется в практике расчета ускорителей для определения устойчивости частиц в фазовом пространстве электромагнитных полей.

Достоинство метода состоит в том, что он позволяет определить область устойчивости работы и выбрать номинальные значения внутренних параметров оптимальным образом.

Существенный недостаток метода заключается в том, что он позволяет провести исследование при одновременном изменении только двух элементов. В реальных условиях погрешность выходного параметра является функцией погрешностей всех внутренних параметров. Поэтому значения погрешностей выходного параметра получаются приближенными.

3. При анализе *методом моментов* постулируют нормальный закон распределения погрешностей внутренних и выходного параметров. Исходными являются характеристики закона распределения внутренних параметров.

Расчет точности по методу моментов сводится к определению среднего значения (математического ожидания) определяемого выходного параметра и его среднего квадратического отклонения или дисперсии

$$\gamma = \varphi(m_{x_1}, m_{x_2}, \dots, m_{x_n}),$$

где m_{x_i} – математическое ожидание внутренних параметров; γ – математическое ожидание выходной функции y , а дисперсия выходной функции определяется следующим образом:

$$Dy = \sum_{i=1}^n (\partial \varphi / \partial x_i)_m \cdot Dx_i,$$

где $\left(\frac{\partial \varphi}{\partial x_i} \right)_m$ – производная функции по i -му параметру в точке математического ожидания m этого параметра.

Этот метод дает точные результаты только для линейных зависимостей и при нормальном распределении погрешностей. Для нелинейных зависимостей даже при нормальном законе распределения имеют место значительные ошибки.

4. *Натурные испытания* – этот эмпирический метод оценки параметров. В качестве модели используется несколько образцов. На каждом образце измеряют уровень параметра. Результаты измерений обрабатываются статистически.

Этот метод требует больших затрат и на этапе проектирования неприменим.

3.4. Метод статистических испытаний (метод Монте-Карло)

Перечисленные ранее методы оценки допустимых отклонений внутренних и внешних параметров от их номинальных значений широко используются в практике проектирования и играют важную роль при предварительной (грубой) оценке допусков.

Наиболее эффективным и современным является метод Монте-Карло, который получил в последнее время очень широкое распространение благодаря широкому внедрению компьютерных технологий, поскольку его применение требует компьютеров очень высокой производительности. Этот метод имеет гораздо более широкие области применения при проектировании (анализе и синтезе математических моделей), нежели расчет допусков, о чем будет сказано позже. Именно с этим методом, прежде всего, связано понятие имитационного моделирования [8].

Наибольшее распространение метод Монте-Карло получил в исследованиях по физике, в теории массового обслуживания, в теории игр и т.п.

Название метода произошло от города Монте-Карло (княжество Монако), известного своими казино, ибо одним из простейших приборов для генерирования случайных чисел (основы метода Монте-Карло) является рулетка.

Официальной датой рождения методов под общим названием Монте-Карло считают 1949 г., когда появилась статья с названием «метод Монте-Карло» (Metropolis N., Ulam S. The Monte Carlo method // I. Amer. Statistical assoc. 1949. 44. № 247. 335–341). Создателями этого метода считают американских математиков Дж. Неймана и С. Ульма. В Советском Союзе первые статьи о методе Монте-Карло были опубликованы в 1955–1956 гг. Возникновение этого метода связано с именами Дж. Неймана, Г. Кана, Э. Ферми и других выдающихся ученых. Все они в 40-х годах работали в Лос-Аламосе над Манхэттенским проектом. Разработки и широкое использование этого метода явилось важным инструментом при реализации проекта. Хотя теоретические основы метода Монте-Карло были известны намного раньше, однако до появления вычислительной техники метод не мог считаться универсальным численным методом ввиду трудоемкости ручного моделирования.

Можно выделить два основных направления применения метода Монте-Карло.

Во-первых, для исследования влияния случайных факторов, естественным образом присутствующих в структуре объекта, учет этих факторов в рамках имитационного моделирования имеет очень важное значение при проектировании.

Во-вторых, этот метод стал активно использоваться для решения детерминированных задач, т.е. задач, модели которых не содержат элемент случайности. В этом проявляется универсальность этого метода. Решение таких задач достигается построением вспомогательных вероятностных моделей, куда в качестве параметров входят подлежащие определению постоянные величины.

В настоящее время созданы вероятностные модели для решения различных математических задач, в их числе вычисление интегралов; решение систем линейных алгебраических уравнений, краевых задач для уравнений эллиптического и параболического типов; обращение матриц; нахождение собственных значений и собственных векторов матриц; вычисление кратных интегралов; решение задач поисковой оптимизации и др.

Поскольку основной идеей при решении детерминированных задач методом Монте-Карло является замена детерминированной задачи эквивалентной статистической, то естественно, что при такой замене вместо точного решения задачи получается приближенное, погрешность которого уменьшается с увеличением числа испытаний.

В качестве примера применения метода Монте-Карло для решения детерминированных задач рассмотрим методику вычисления интегралов (площадей, объемов) [9].

Допустим, требуется вычислить определенный интеграл функции $y = f(x)$

$$J = \int_0^1 f(x) dx .$$

Если пределы изменения функции и аргумента – произвольные, то путем соответствующей нормировки всегда можно добиться, чтобы величина x и функция $f(x)$ изменялись в пределах $[0, 1]$ (рис. 3.1) при $0 \leq x \leq 1$.

Практически требуется вычислить площадь под кривой, изображенной на рисунке (площадь ограничена кривой $y = f(x)$, осью абсцисс и прямыми $x = 1$ и $y = 1$).

Для решения этой задачи необходимо иметь два независимых датчика случайных чисел, которые выдают две последовательности случайных чисел x и y , равномерно распределенных в интервале

$[0, 1]$. Пусть в квадрат 1×1 попала случайная точка O с координатами $[\zeta, \eta]$. Если для этой точки выполняется условие

$$f(\zeta) > \eta,$$

то точка попала под кривую $y = f(x)$.

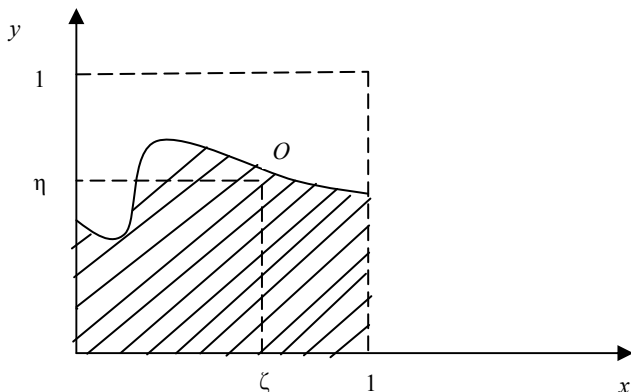


Рис. 3.1. Пояснение к процессу вычисления интеграла по методу Монте-Карло

Продлав этот эксперимент N раз и определив количество точек L , для которых выполняется это условие, получим

$$\frac{L}{N} \approx P = \int_0^1 f(x) dx.$$

Таким образом, задача решена, хотя, в принципе, при решении этой задачи можно обойтись одним датчиком случайных чисел.

Однако наибольшее значение метод Монте-Карло приобрел в САПР при исследовании влияния случайных факторов (временных или пространственных), непременно имеющих место при работе любого изделия. Он является основным методом многовариантного анализа в настоящее время.

Рассмотрим несколько примеров использования этого метода в рамках имитационного моделирования [9].

1. *Определение допусков на номинальные параметры элементов электронных схем, входящих в состав разнообразных приборов (измерителей, генераторов, систем управления и т.п.).* Предположим, что номинальное значение выходного параметра U требуется

поддерживать с точностью не хуже 1 % от его номинала. Значения U можно рассчитать, зная параметры их элементов, используя модели теории электрических цепей, о которых говорилось ранее:

$$U = f(R_1, R_2, \dots, R_n, C_1, C_2, \dots, C_n, \dots).$$

Здесь U может представлять собой выходной импульс с определенными параметрами (амплитудой, длительностью, требованиями к фронтам и т.п.); $R_1, R_2, \dots, R_n, C_1, C_2, \dots, C_n$ – сопротивления, конденсаторы и другие электрорадиотехнические элементы.

Возникают вопросы: как могут повлиять на выходные характеристики импульса отклонения в номинальных значениях параметров элементов схемы и какие, например, нужно брать сопротивления с 5- или 10-процентным разбросом от указанного номинала?

При большом количестве элементов упомянутые ранее методы (наихудшего случая, метод моментов) дают сильно завышенные требования к допускам. Вычислить аналитически распределение U (учитывая, что разброс в серийных элементах схемы, выпускаемых миллионными тиражами, подчиняется нормальному закону) при мало-мальски сложной функции $f(x)$ невозможно.

В связи с этим единственным приемлемым методом для решения этой задачи является метод Монте-Карло. В этом случае поступают следующим образом. Считают, что элементы, например сопротивления R , – случайные величины, имеющие распределения Гаусса с математическим ожиданием M_R , равным номинальному значению, а среднеквадратическое отклонение связано с точностью изготовления каждого параметра (допуском, указанным на рассматриваемый элемент) следующим образом: $\Delta = 3\sigma$.

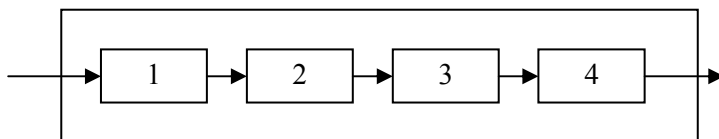
Алгоритм расчета допусков оказывается очень простым: для каждого элемента разыгрывается значение параметра, затем вычисляется значение U_1 . Повторив этот опыт N раз и получив значения $U_1, U_2, \dots, U_N$, можно построить статистическое распределение (гистограмму) и приближенно рассчитать:

$$MU \approx \frac{1}{N} \sum_{j=1}^N U_j \text{ – математическое ожидание;}$$

$$DU \approx \frac{1}{N-1} \left[\sum_{j=1}^N U_j^2 - \frac{1}{N} \left(\sum_{j=1}^N U_j \right)^2 \right] \text{ – дисперсия.}$$

2. *Расчет надежности работы изделия.* Пусть хотим оценить время безотказной работы изделия, предполагая, что известны характеристики безотказной работы каждого элемента.

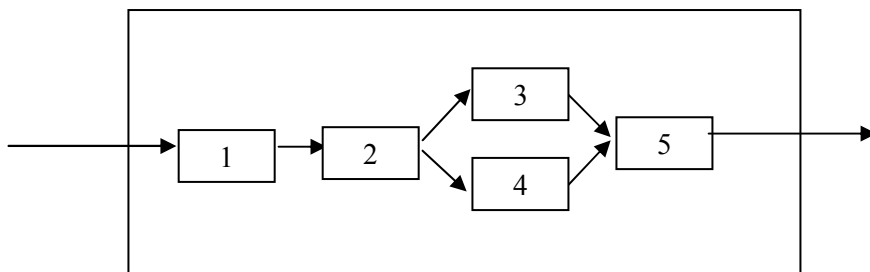
Если считать, что время безотказной работы каждого элемента t_n – фиксированная величина, то вычислить время безотказной работы t изделия не составляет труда. Например, рассмотрим рисунок:



На рисунке схематически изображено изделие, в котором выход из строя одного элемента влечет за собой выход из строя всего изделия:

$$t = \min(t_1, t_2, t_3, t_4).$$

Теперь перейдем к рисунку:



На нем схематически изображено изделие, в котором один из элементов дублирован:

$$t = \min(t_1; t_2; \max(t_3, t_4); t_5),$$

так как если, например, элемент 3 выйдет из строя, то изделие будет продолжать работать на элементе 4.

В действительности, время безотказной работы любого элемента представляет случайную величину Q_k .

Когда говорим, что срок службы электрической лампочки 1000 ч, то это лишь среднее значение M_Q величины Q . Всем известно, что одна лампочка может перегореть быстрее, а другая (та же) горит дольше.

Если известна плотность распределения вероятности Q_k для каждого из элементов изделия, то можно значительно точнее решить эту задачу, используя метод Монте-Карло.

В самом деле для каждого элемента можно разыграть значение случайной величины θ_k , пусть это будет t_k , затем по соответствующей из вышеперечисленных формул можно вычислить t случайной величины Q . Повторив этот опыт многократно (N раз), можно считать, что

$$M_Q \approx \frac{1}{N} \sum_{j=1}^N t_j,$$

где t_j значение t , полученное в j -м опыте.

Необходимо отметить, что вопрос о распределении времени жизни Q_k для отдельных элементов не так уж прост: для наиболее долговечных элементов организация эксперимента (по нахождению закона распределения) затруднительна, так как нужно дожидаться, пока выйдет из строя достаточно много элементов.

3. *Расчет прохождения нейтронов сквозь пластинку.* Наибольшее распространение метод Монте-Карло получил в нейтронной физике. Достаточно напомнить, что критическая масса урана в США была определена путем статистического моделирования на электронной вычислительной машине «Эниак». Далее будет рассмотрена упрощенная задача о прохождении нейтронов через пластинку [9]. На однородную бесконечную пластинку толщиной h падает под углом 90° поток нейтронов с энергией E_0 . При прохождении нейтронов через вещество, из которого состоит пластинка, происходят столкновения с атомами этого вещества. Для простоты предполагается, что возможны два вида удара: идеально упругий и неупругий, когда нейтрон поглощается. В первом случае считается, что нейтрон рассеивается равновероятно в любом направлении, и энергия его сохраняется. Могут представиться три случая, изображенные на рис. 3.2. Ставится задача определить вероятности этих трех событий: p^+ , p^0 и p^- .

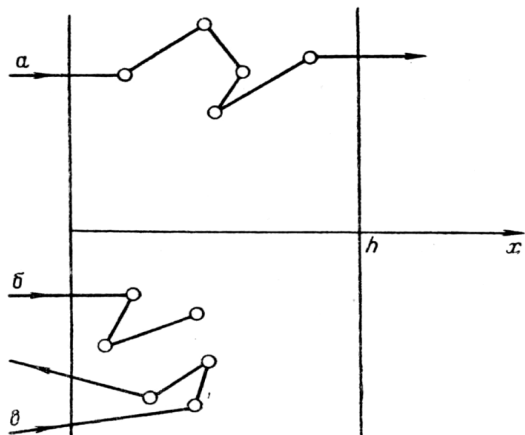


Рис. 3.2. Возможные случаи при прохождении нейтронов через пластинку: a – прохождение сквозь пластинку; $б$ – поглощение; $в$ – отражение

В нейтронной физике вводятся характеристики взаимодействия нейтронов с веществом, называемые сечением поглощения Σ_c и сечением рассеивания Σ_s . Эти величины определяют вероятность поглощения нейтрона Σ_c/Σ и вероятность его рассеивания Σ_s/Σ при его столкновении с атомом вещества. Здесь $\Sigma = \Sigma_c + \Sigma_s$ и называется *полным сечением*.

Считается, что длина свободного пробега нейтрона λ (расстояние от столкновения до столкновения) является положительной случайной величиной с экспоненциальной плотностью распределения вероятности

$$p_\lambda(\lambda) = \Sigma e^{-\Sigma\lambda}.$$

Математическое ожидание свободного пробега λ обратно пропорционально полному сечению:

$$M[\lambda] = \frac{1}{\Sigma}.$$

Для получения случайной величины λ из случайной величины γ , имеющей равномерный закон распределения, необходимо осуществить операцию разыгрывания в соответствии с формулой:

$$\int_0^{\lambda} p(x) dx = \gamma$$

или

$$\int_0^{\lambda} \Sigma e^{-\Sigma x} dx = \gamma.$$

После взятия интеграла в левой части получим окончательно формулу для разыгрывания в виде

$$\lambda = \frac{1}{\Sigma} \ln(1 - \gamma).$$

Так как величина γ – положительная и равномерно распределенная, то формулу можно переписать:

$$\lambda = -\frac{1}{\Sigma} \ln \gamma,$$

имея в виду, что закон распределения величины $1 - \gamma$ также равномерный в интервале $[0, 1]$.

Теперь необходимо выяснить, как выбирать направление нейтрона после удара. Оно может быть определено углом φ между осью OX и направлением движения нейтрона. Покажем, что условие «равно вероятности любого направления» эквивалентно равномерности распределения величины $\mu = \cos \varphi$ в интервале $(-1, 1)$. Переходя к пространственной задаче, условие «равно вероятности» можно сформулировать как требование того, что точка A на сфере единичного радиуса с центром в точке столкновения нейтрона с атомом окажется равновероятным образом в любом элементе поверхности сферы dS с вероятностью $\frac{dS}{4\pi}$.

В данном случае точка A может рассматриваться как точка пересечения траектории нейтрона после столкновения. Выбираем сферические координаты (φ, ψ) с полярной осью OX , причем $0 \leq \psi \leq \pi$, $0 \leq \varphi \leq 2\pi$. Очевидно, что $dS = \sin \psi d\varphi d\psi$. Используя это соотношение и условие независимости координат φ и ψ , можем написать для плотности распределения точки A на сфере с координатами φ и ψ выражение:

$$p(\varphi, \psi) = p_\psi(\varphi) \frac{\sin \psi}{4\pi}.$$

Интегрируя это выражение по φ от 0 до 2π и учитывая, что

$$\int_0^{2\pi} p_\psi(\varphi) d\varphi = 1,$$

получим

$$\int_0^{2\pi} p(\varphi, \psi) d\varphi = \frac{\sin \psi}{2}.$$

Имеем $p_\psi(\varphi) = \frac{1}{2\pi}$, т.е. величина ψ имеет равномерный закон распределения в интервале $(0, 2\pi)$, и для разыгрывания следует использовать формулу:

$$\psi = 2\pi\gamma.$$

Соответствующая формула разыгрывания для величины φ получается из соотношения:

$$\frac{1}{2} \int_0^\varphi \sin x dx = \gamma$$

Отсюда

$$\mu = \cos \varphi - \gamma.$$

Из этой формулы следует, что при изменении γ в интервале $[0, 1]$ μ изменяется в интервале $[-1, 1]$. Кроме того, формула розыгрыша случайной величины μ , равномерно распределенной в интервале $[-1, 1]$, может быть получена из соотношения:

$$\int_{-1}^\mu \frac{dx}{2} = \gamma,$$

следовательно, $\mu = 2\gamma - 1$. Эта формула получается заменой γ на $1 - \gamma$, а обе эти величины имеют одинаковое распределение. Таким образом, для разыгрывания угла φ необходимо использовать формулу, примененную для разыгрывания $\cos \varphi$. Тогда $\lambda_k = -(1/\Sigma) \ln \gamma$, и вычислим абсциссу следующего столкновения

$$x_{k+1} = x_k + \lambda_k \mu_k.$$

Перейдем непосредственно к построению схемы моделирования. Обозначим абсциссы двух последовательных столкновений нейтрона с атомами через x_k и $x_k + x$. Разыграем длину свободного пробега после k -го столкновения по формуле:

$$\lambda_k = -(1/\Sigma) \ln \gamma,$$

где $\mu_k = \cos \varphi_k$ – направление движения нейтрона после k -го столкновения. Далее проверим условие прохождения нейтрона сквозь пластинку: $x_{k+1} > h$. При удовлетворении этого условия расчет траектории заканчивается и добавляется единица к счетчику прошедших частиц. Если это условие не выполняется, то следует разыграть $k + 1$ столкновений. Для этого выбираем очередное значение γ и проверяем условие поглощения: $\gamma < \frac{\Sigma_\gamma}{\Sigma}$.

При выполнении этого условия расчет траектории заканчивается и добавляется единица к счетчику поглощенных частиц. В противном случае электрон испытал рассеивание в точке $x_k + 1$, и разыгрывается новое направление скорости движения нейтрона по формуле:

$$\mu_{k+1} = 2\gamma - 1.$$

По существу, для расчета k -го шага требуется три значения γ . Начальные значения для всех траекторий можно выбрать следующие:

$$x_0 = 0, \quad \mu_0 = 1.$$

В результате N -кратного расчета траекторий определяются числа нейтронов: N^+ -прошедших сквозь пластинку, N^- -отразившихся от нее, N^0 -поглощенных веществом пластинки. Соответственно, вероятности определяются по формулам:

$$p^+ \approx \frac{N^+}{N}, \quad p^- \approx \frac{N^-}{N}, \quad p^0 \approx \frac{N^0}{N}.$$

Блок-схема программы приведена на рис. 3.3, где j – номер траектории ($j = 1, \dots, N$) и k – номер столкновения. На этой схеме после блока, проверяющего некоторое условие, имеются два выхода, обозначенные 1, если условие выполняется, и 0, если условие не выполняется.

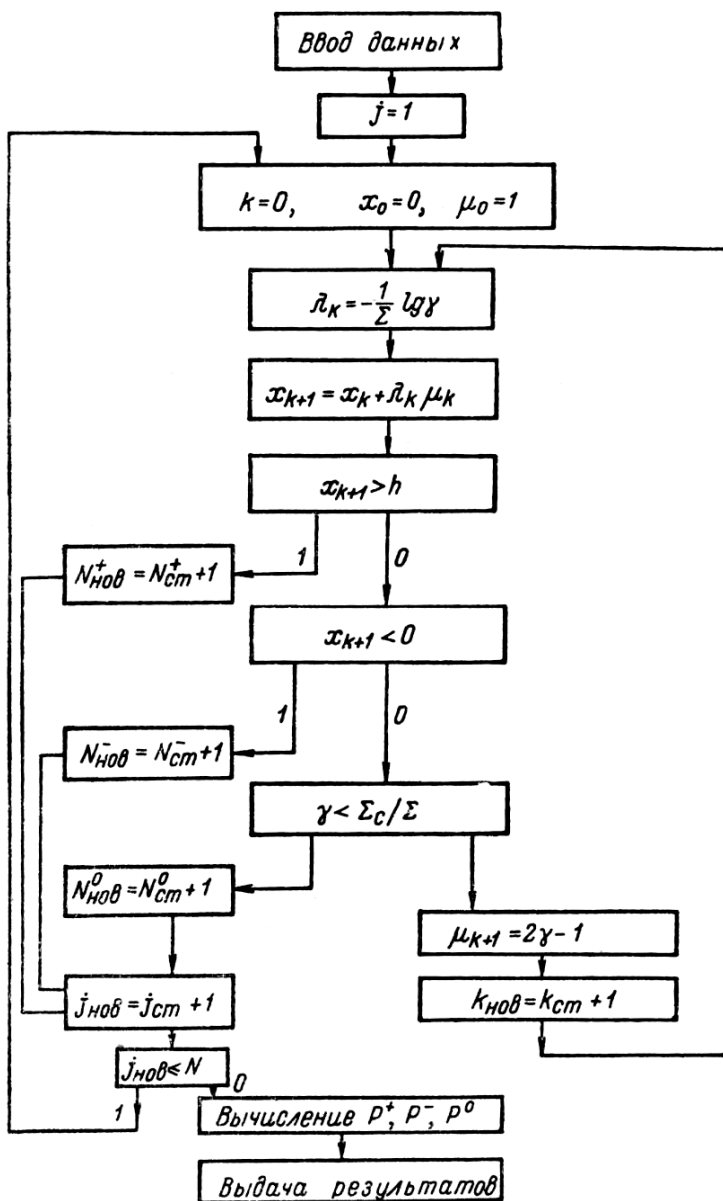


Рис. 3.3. Блок-схема программы

В приведенных примерах метод Монте-Карло применяется в задачах самой разной направленности, однако во всех этих задачах можно выделить общие характерные черты моделирования, единые для любого приложения, связанного с имитационным моделированием.

Обобщенная структура моделирования методом Монте-Карло чрезвычайно проста и может быть представлена в виде следующих схем блоков (рис. 3.4.).

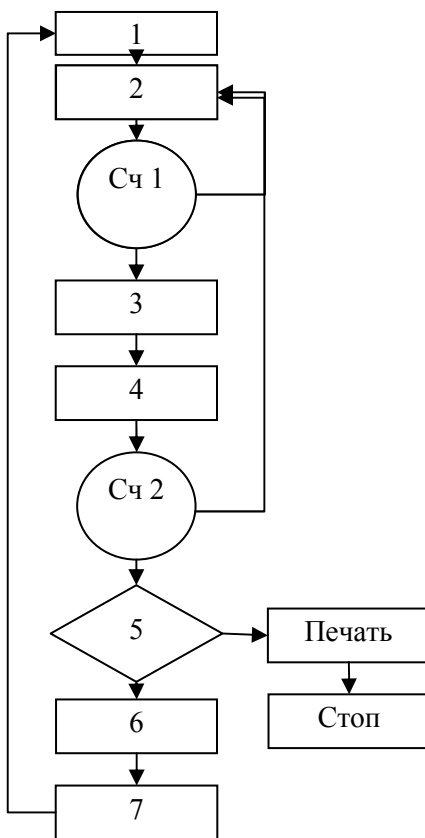


Рис. 3.4. Обобщенная блок-схема алгоритма метода Монте-Карло

На рисунке даны следующие обозначения:

блок 1 – ввод исходных данных – номинальных значений внутренних и внешних параметров, дисперсии случайных величин (законов распределений), числа испытаний;

блок 2 – генератор случайных (псевдослучайных, квазислучайных) чисел;

Сч 1 – счетчик, определяющий количество случайных параметров, необходимых при моделировании;

блок 3 – расчет стационарной модели конкретного приложения (аналитика, системы линейных дифференциальных уравнений, уравнения в частных производных и т.п.);

блок 4 – блок текущего построения гистограммы (статистического распределения);

Сч 2 – счетчик общего количества испытаний (задаваемого в блоке 1);

блок 5 – блок сравнения, насколько удовлетворительно знание общего количества испытаний с точки зрения точности получаемых результатов, а также насколько удовлетворительны заданные требования к внутренним и внешним параметрам системы;

блок 6 – определение параметров, для которых заданные требования к допустимым отклонениям их от номинальных значений нуждаются в коррекции;

блок 7 формирует команду на изменение входных данных (блок 1) – изменение количества испытаний, изменение дисперсии распределения параметра, обладающего наибольшим весом в данный момент и т.п.

Однако не все так просто, реализация рассмотренного алгоритма предполагает решение ряда вероятностно-статистических задач.

1. Получение последовательности случайных чисел с различными законами распределения. Успех и точность статистического моделирования зависят в основном от двух факторов: качества последовательности случайных чисел и выбора оптимального алгоритма моделирования.

Исходным материалом для формирования случайных последовательностей чисел с различными законами распределения служат случайные числа x_i , имеющие равномерное распределение в интервале $[0, 1]$. Таким образом, задача получения случайных чисел обычно разбивается на две. Вначале получают последовательность случайных чисел, имеющих равномерное распределение. Затем одним из способов преобразования получают из последовательности равномерно распределенных чисел последовательность с нужным распределением.

Другими словами, случайные числа x_i , как возможные значения случайной величины ζ , имеющей равномерное распределение в интервале $[0, 1]$ могут быть преобразованы в возможные значения y_i случайной величины η , закон распределения которой требуется.

Существует два основных пути такого преобразования случайных чисел. Один из них может быть назван прямым. Он состоит в реализации некоторой операции над числом x_i , формирующей число y_i , имеющее требуемый закон распределения. Другой путь основывается на моделировании условий, соответствующей предельной теоремы теории вероятностей.

Рассмотрим сущность первого пути. Идея построения интересующей нас операции вытекает из следующей теоремы теории вероятностей [7].

Теорема. Если случайная величина η имеет плотность распределения $f(y)$, то распределение случайной величины

$$\varsigma = \int_0^{\eta} f(y) dy$$

является равномерным в интервале $[0, 1]$.

На основании этой теоремы можно прийти к следующему правилу.

Чтобы получить число, принадлежащее совокупности случайных чисел y_i , имеющей функцию плотности распределения $f(y)$, необходимо разрешить относительно y_i уравнение

$$\int_{-\infty}^{y_i} f(y) dy = x_i.$$

Варианты реализации этого уравнения представлены в [7–9].

Геометрическая интерпретация преобразования и графический способ получения случайных чисел с любым законом распределения можно пояснить следующим образом.

На рис. 3.5 $F(x)$ представлена необходимая нам интегральная функция распределения случайной величины x . Если y является функцией x , т.е. $y = F(x)$, то $0 \leq y \leq 1$. Поэтому для получения последовательности случайных чисел, имеющих данную (интегральную) функцию распределения $F(x)$, необходимо каждое число y с

выхода генератора случайных чисел, который формирует числа с равномерным законом распределения в интервале $[0, 1]$, подать на нелинейное устройство (аналоговое или цифровое), в котором реализуется функция, обратная $F(x)$ т.е.

$$x = F^{-1}(y).$$

Реализуемая таким образом величина x будет иметь функцию распределения $F(x)$. Эта процедура может быть использована для графического способа получения случайных величин.

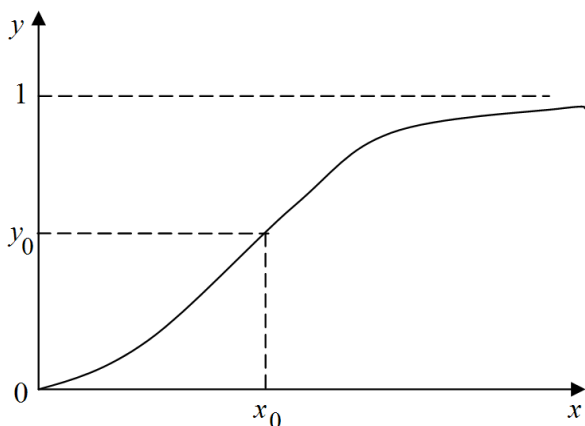


Рис. 3.5. Пояснение к процессу получения случайных чисел с любым законом распределения

Так, для получения случайной величины x (см. рис. 3.5) берется число y из последовательности равномерного распределения в интервале $[0, 1]$, которое откладывается на оси y , а на оси x считывается соответствующее число x_0 . Повторив неоднократно эту процедуру, получим набор случайных чисел, имеющих закон распределения $F(x)$.

Таким образом, получение случайных чисел различных распределений связано с необходимостью получения случайных равномерно распределенных чисел в интервале $[0, 1]$. Без преувеличения можно сказать, что наличие простых и экономных способов образования последовательности случайных чисел в интервале $[0, 1]$ во

многом определяет возможность практического использования метода Монте-Карло.

Напомним основные свойства равномерного распределения. Непрерывная случайная величина x имеет равномерные распределения в интервале $[a, b]$, если ее функция плотности равна

$$f(y) = \begin{cases} 0 & \text{при } x < a; \\ b - a & \text{при } a \leq x \leq b; \\ 0 & \text{при } x > b. \end{cases}$$

Функция распределения случайной величины η имеет вид

$$F(x) = \begin{cases} 0 & \text{при } x < a; \\ \frac{y-a}{b-a} & \text{при } a \leq x \leq b; \\ 1 & \text{при } y > b. \end{cases}$$

Математическое ожидание $M[\eta]$ и среднее квадратичное отклонение σ_η , соответственно, равны

$$M[x] = \frac{a+b}{2}, \quad \sigma = \frac{b-a}{2\sqrt{3}}.$$

В частном случае равномерного распределения в отрезке $[0, 1]$ случайная величина η^* имеет функцию плотности

$$f(x) = \begin{cases} 1 & \text{при } 0 \leq x \leq 1; \\ 0 & \text{вне этого интервала;} \end{cases}$$

функцию распределения

$$F(x) = \begin{cases} 0 & \text{при } x < 0; \\ y & \text{при } 0 \leq x \leq 1; \\ 1 & \text{при } y > 1; \end{cases}$$

а математическое ожидание и среднее квадратичное отклонение, соответственно, равны

$$M[x^*] = \frac{1}{2}, \quad \sigma_x^* = \frac{1}{2\sqrt{3}}.$$

2. *Получение последовательности случайных чисел, равномерно распределенных в интервале [0, 1].* Исторически сложилось два способа получения случайных чисел, равномерно распределенных в интервале [0, 1]:

1) таблица случайных чисел, которая практически вышла из употребления в настоящее время.

2) генераторы случайных чисел, которые, в свою очередь, подразделяются на:

- генераторы, использующие физический способ генерирования (датчики истинно случайных чисел);
- генераторы, использующие алгоритмический способ получения, датчики так называемых псевдослучайных чисел.

При физическом способе получения случайных чисел один из методов состоит в формировании для компьютера дискретной случайной величины, которая может принимать только два значения 0 или 1 с вероятностями

$$P(z_i = 0) = \frac{1}{2};$$

$$P(z_i = 1) = \frac{1}{2}.$$

Далее, если рассмотреть бесконечную последовательность $z_1, z_2, z_3, \dots$ как значения числа ζ^* вида

$$\zeta^* = z_1 \cdot 2^{-1} + z_2 \cdot 2^{-2} + \dots + z_n \cdot 2^{-n} + \dots,$$

то можно доказать, что случайная величина ζ^* , заключенная в интервале [0, 1], имеет равномерный закон распределения [8]. Отсюда вытекает способ формирования равномерно распределенной случайной величины. Нужно взять бесконечную последовательность независимых случайных величин z_i и считать их двоичными знаками некоторого числа ζ .

Строго говоря, на цифровой вычислительной машине получить последовательность возможных значений физической величины с равномерным распределением не представляется возможным в силу ограниченного количества двоичных разрядов машинного слова или, другими словами, членов ряда числа ζ^* .

Предположим, что в выбранном нами компьютере числа представляются k двоичными разрядами. Тогда количество несовпа-

дающих между собой чисел, каждое из которых можно записать в k разрядную ячейку компьютера, равно 2^k . Поэтому приходится вместо непрерывной совокупности случайных чисел с равномерным распределением в качестве исходной использовать дискретную совокупность 2^k чисел с одинаковыми вероятностями появления любого из них.

Такое распределение иногда называют *квазиравномерным*, а генераторы, использующие физические способы генерирования, называют *генераторами истинно случайных чисел*. Однако следует отметить, что расхождения в параметрах распределений (равномерного и квазиравномерного) даже на 16-разрядной сетке сравнительно небольшое.

Если учесть, что оценить две основные характеристики равномерного распределения математического ожидания M и дисперсию σ можно, используя следующее соотношение [8]:

$$M = \frac{1}{2} \left(1 - \frac{1}{k} \right), \quad \sigma = \frac{1}{2\sqrt{3}} \sqrt{1 - \frac{1}{2}} k,$$

то можно получить табл. 3.1, в которой приводятся измерения значения дисперсии квазиравномерного распределения σ_ζ в зависимости от разрядности сетки и относительная ошибка этого параметра (по отношению к точному значению σ).

Таблица 3.1

k	2	3	5	10	15
σ_ζ	0,3727	0,3274	0,2979	0,2889	0,2887
σ_ζ/σ	1,290	1,140	1,030	1,001	1,000

Таким образом, уже на 15-ти разрядах ошибка в оценке дисперсии наблюдается в пятом знаке. Ошибка оценки математического ожидания и того меньше.

На первых вычислительных машинах в качестве генераторов истинно случайных чисел применялись специальные приставки, наиболее часто использующие либо радиоактивные источники, в которых за равные интервалы времени регистрировалось количество испускаемых частиц (четное $z=0$, нечетное $z=1$); либо шумы электронных ламп («белый» шум), в которых за равные промежутки времени регистрировались превышения колебаний напряжения

на аноде над номинальным значением и также четные превышения оценивались как $z = 0$, а нечетные как $z = 1$. В современных микропроцессорах «привязка» производится к внутренним процессам кристалла. Так, в процессорах x86 сообщалось, что уже у МП Pentium II появился генератор истинно случайных чисел. Основным недостатком генераторов истинно случайных чисел считается невозможность воспроизведения одинаковых последовательностей чисел, т.е. повторяемости одной и той же последовательности, что бывает очень важным свойством генератора в процессе проектирования. Второй способ получения случайных последовательностей связан с генерацией так называемых псевдослучайных чисел непосредственно на вычислительной машине с помощью специальных алгоритмов.

Первый алгоритм для получения псевдослучайных чисел был предложен Дж. Нейманом. Он назывался методом срединного квадрата. Алгоритм этого метода чрезвычайно прост. Берется четырехзначное число, возводится в квадрат, выбираются четыре цифры из середины возведенного числа и эта процедура повторяется многократно. В качестве случайных чисел используются полученные числа, умноженные на 10^{-4} , т.е. получаются числа в интервале $[0, 1]$. Но этот алгоритм не оправдал себя, поскольку алгоритм выдавал неравномерно большее число малых значений в получаемой последовательности.

В настоящее время разработано большое количество генераторов псевдослучайных чисел. Практически во всех наиболее распространенных языках программирования наряду с библиотеками элементарных функций включаются генераторы псевдослучайных чисел.

Однако наряду с очевидными достоинствами этих генераторов – возможность многократного воспроизведения одних и тех же последовательностей чисел, простота алгоритмов и способа их реализации, основным и весьма серьезным недостатком такого способа получения случайных чисел является ограниченность «запаса» псевдослучайных чисел. Имеется в виду повторяемость или цикличность (периодичность) в последовательности случайных чисел, которая наступает значительно раньше, чем генераторы истинно случайных чисел заполняют разрядную сетку компьютера. Точные, аналитические методы оценки периодичности, как правило, отсутствуют, тем более что большинство генераторов используют эвристические ал-

горитмы. В связи с этим большое значение приобретают методы экспериментальной проверки качества этих алгоритмов. Хотя проверять качество генераторов истинно случайных чисел также целесообразно из-за возможных сбоев в аппаратуре. Этот вопрос будет рассмотрен в дальнейшем. Предварительно необходимо коснуться методов статистической обработки получаемых случайных последовательностей, которая предваряет проверку их качества.

3. *Понятие статистического ряда и гистограммы* [7]. Наиболее полное представление о последовательности случайных чисел дают их плотности распределения. Описание статистических последовательностей представляется статистическим рядом, а графически в виде гистограммы.

Предположим, что в нашем распоряжении результаты генерации случайной величины X . Разделим весь диапазон наблюдаемых значений X на интервалы (или разряды) и подсчитаем количество значений m_i , приходящее на каждый интервал (разряд). Это число разделим на общее число генерируемых чисел n и найдем частоту (вероятность), соответствующую данному разряду:

$$P_i^* = \frac{m_i}{n}.$$

Сумма частот должна быть равна единице, а их последовательность представляется в виде табл. 3.2, называемой статистическим рядом.

Таблица 3.2

i	X_1, x_2	x_2, x_3	...	x_{i-1}, x_i	...	x_k, x_{k+1}
P_i	P_1	P_2	...	P_i^*	...	P_k^*

Если случайное число попадает на границу интервала, то можно добавлять в каждый интервал по 1/2.

Число разрядов, из которых следует группировать статистический ряд, не должно быть слишком большим (при этом проявляются незакономерные колебания), при слишком малом числе разрядов распределения оцениваются слишком грубо.

Практика показывает, что в большинстве случаев рационально выбирать число разрядов 10–20. Размер случайной последовательности (количество случайных чисел) следует подбирать таким образом, чтобы в каждый подынтервал попадало не менее десяти чисел.

Статистический ряд оформляется графически в виде так называемой гистограммы (иногда в виде спектра или полигона) рис. 3.6. По оси абсцисс откладывают разряды, а по каждому из разрядов, как на основании строится прямоугольник, площадь которого равна частоте попадания случайной величины в данный разряд. Для построения гистограммы необходимо частоту каждого разряда разделить на его длину, а полученное число взять в качестве высоты прямоугольника. Из способа построения гистограммы следует, что полная площадь ее равна единицы.

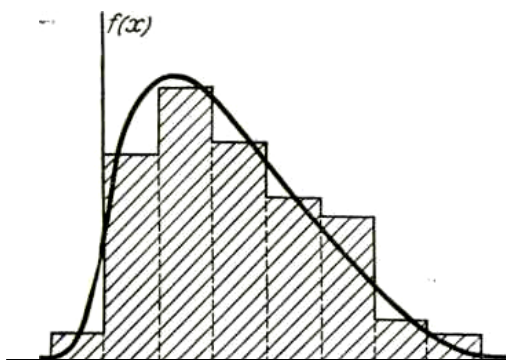


Рис. 3.6. Гистограмма случайной величины распределений

Гистограмма является основным объектом оценки качества генераторов случайных (или псевдослучайных) последовательностей чисел, при использовании критериев согласия.

4. *Критерии согласия* [7]. Вопрос о качестве того или иного способа генерации случайных последовательностей равномерно распределенных чисел является одним из основных (если не основным) при проведении статистического моделирования.

Естественно, возникает вопрос: объясняются ли естественные расхождения между теоретическим распределением и полученной (статистической) гистограммой случайными обстоятельствами, связанными с ограниченным числом испытаний или они являются существенными и связаны с низким качеством способа генерации? Для ответа на этот вопрос служат так называемые «критерии согласия». Идея применения критериев согласия заключается в следующем.

На основании данного статистического материала нам предстоит проверить гипотезу H , состоящую в том, что случайная величина X подчиняется некоторому определенному закону распределения. Этот закон может быть задан в той или иной форме: например, в виде функции распределения $F(x)$ или в виде плотности распределения $f(x)$, или же в виде совокупности вероятностей p_i , где p_i – вероятность того, что величина X попадет в пределы i -го разряда.

Так как из этих форм функция распределения $F(x)$ является наиболее общей и определяет собой любую другую, будем формулировать гипотезу H , как состоящую в том, что величина X имеет функцию распределения $F(x)$.

Для того чтобы принять или опровергнуть гипотезу H , рассмотрим некоторую величину U , характеризующую степень расхождения теоретического и статистического распределений. Величина U может быть выбрана различными способами; например, в качестве U можно взять сумму квадратов отклонений теоретических вероятностей P_i от соответствующих частот p_i^* или же сумму тех же квадратов с некоторыми коэффициентами («весами»), или же максимальное отклонение статистической функции распределения $F^*(x)$ от теоретической $F(x)$ и т.д. Допустим, что величина U выбрана тем или иным способом. Очевидно, это есть некоторая случайная величина. Закон распределения этой случайной величины зависит от закона распределения случайной величины X , над которой производились опыты, и от числа опытов n . Если гипотеза H верна, то закон распределения величины U определяется законом распределения величины X (функцией $F(x)$) и числом n .

Допустим, что этот закон распределения нам известен. В результате данной серии опытов обнаружено, что выбранная нами мера расхождения U приняла некоторое значение u . Спрашивается, можно ли объяснить это случайными причинами или же это расхождение слишком велико и указывает на наличие существенной разницы между теоретическим и статистическим распределениями и, следовательно, на непригодность гипотезы H ? Для ответа на этот вопрос предположим, что гипотеза H верна, и вычислим в этом предположении вероятность того, что за счет случайных причин, связанных с недостаточным объемом опытного материала, мера

расхождения U окажется не меньше, чем наблюдаемое нами в опыте значение u , т.е. вычислим вероятность события: $U \geq u$.

Если эта вероятность весьма мала, то гипотезу H следует отвергнуть как маловероятную; если же эта вероятность значительна, следует признать, что экспериментальные данные не противоречат гипотезе H .

Возникает вопрос о том, каким же способом следует выбирать меру расхождения U . Оказывается, что при некоторых способах ее выбора закон распределения величины U обладает весьма простыми свойствами и при достаточно большом n практически не зависит от функции $F(x)$. Именно такими мерами расхождения и пользуются в математической статистике в качестве критериев согласия.

Рассмотрим один из наиболее часто применяемых критериев согласия – так называемый «критерий χ^2 » Пирсона.

Предположим, что произведено n независимых опытов, в каждом из которых случайная величина X приняла определенное значение. Результаты опытов сведены в k разрядов и оформлены в виде статистического ряда, о котором говорилось ранее (табл. 3.3).

Таблица 3.3

I_i	$x_1; x_2$	$x_2; x_3$	...	$x_k; x_{k+1}$
p_i^*	p_1^*	p_2^*	...	p_k^*

Требуется проверить, согласуются ли экспериментальные данные с гипотезой о том, что случайная величина X имеет данный закон распределения (заданный функцией распределения $F(x)$ или плотностью $f(x)$, см. рис. 3.6). Назовем этот закон распределения *теоретическим*.

Зная теоретический закон распределения, можно найти теоретические вероятности попадания случайной величины в каждый из разрядов:

$$P_1, P_1, P_1, \dots, P_k.$$

Проверяя согласованность теоретического и статистического распределений, будем исходить из расхождений между теоретическими вероятностями p_i и наблюдаемыми частотами p_i^* . Естест-

венно выбрать в качестве меры расхождения между теоретическим и статистическим распределениями сумму квадратов отклонений $(p_i^*, \dots, p_i)$, взятых с некоторыми «весами» c_i :

$$U = \sum_{i=1}^k c_i (p_i^* - p_i)^2.$$

Коэффициенты c_i («веса» разрядов) вводятся потому, что в общем случае отклонения, относящиеся к различным разрядам, нельзя считать равноправными по значимости. Действительно, одно и то же по абсолютной величине отклонение $p^* - p_i$ может быть мало-значительным, если сама вероятность p_i велика, и очень заметным, если она мала. Поэтому естественно «веса» взять обратно пропорциональными вероятностям разрядов p_i . Далее возникает вопрос о том, как выбрать коэффициент пропорциональности.

К. Пирсон показал, что если положить

$$c_i = \frac{n}{p_i},$$

то при больших n закон распределения величины U обладает весьма простыми свойствами: он практически не зависит от функции распределения $F(x)$ и от числа опытов n , а зависит только от числа разрядов k , а именно: этот закон при увеличении n приближается к так называемому распределению χ^2 .

* Распределением χ^2 с r степенями свободы называется распределение суммы квадратов r независимых случайных величин, каждая из которых подчинена нормальному закону с математическим ожиданием, равным нулю, и дисперсией, равной единице. Это распределение характеризуется плотностью

$$k_r(u) = \begin{cases} \frac{1}{2^{\frac{r}{2}} \Gamma\left(\frac{r}{2}\right)} u^{\frac{r}{2}-1} e^{-\frac{u}{2}} & \text{при } u > 0; \\ 0 & \text{при } u < 0, \end{cases}$$

где $\Gamma(\alpha) = \int_0^{\infty} t^{\alpha-1} e^{-t} dt$ — неизвестная гамма-функция.

При таком выборе коэффициентов c_i мера расхождения обычно обозначается χ^2 :

$$\chi^2 = n \sum_{i=1}^k \frac{(p_i^* - p_i)^2}{p_i}.$$

Для удобства вычислений (чтобы не иметь дело с дробными величинами с большим числом нулей) можно ввести n под знак суммы и, учитывая, что $p_i^* = \frac{m_i}{n}$, где m_i — число значений в i -м разряде, привести предыдущую формулу к виду

$$U = \chi^2 = \sum_{i=1}^k \frac{(m_i - np_i)^2}{np_i}.$$

Распределение χ^2 зависит от параметра r , называемого числом «степеней свободы» r , которое равно числу разрядов k минус число независимых условий («связей»), наложенных на частоты p_i^* .

Примерами таких условий могут быть

$$\sum_{i=1}^k p_i^* = 1,$$

если мы требуем только того, чтобы сумма частот была равна единице (это требование накладывается во всех случаях);

$$\sum_{i=1}^k x_i p_i^* = m_x.$$

Если мы подбираем теоретическое распределение с тем условием, чтобы совпадали теоретическое и статическое средние значения;

$$\sum_{i=1}^k (x_i - m_x^*)^2 p_i^* = D_x,$$

если требуем, кроме того, совпадения теоретической и статистической дисперсий и т.д.

Для распределения χ^2 составлены специальные таблицы. Пользуясь этими таблицами, можно для каждого значения χ^2 и числа степеней свободы r найти вероятность p того, что величина, распределенная по закону χ^2 , превзойдет это значение. В табл. 3.4 входами являются значение вероятности p и чисел степеней свободы r .

Числа, стоящие в таблице, представляют собой соответствующие значения χ^2 .

Таблица 3.4

λ	$p(\lambda)$	λ	$p(\lambda)$	λ	$p(\lambda)$
0,0	1,000	0,7	0,771	1,4	0,040
0,1	1,000	0,8	0,544	1,5	0,022
0,2	1,000	0,9	0,393	1,6	0,012
0,3	1,000	1,0	0,270	1,7	0,006
0,4	0,997	1,1	0,178	1,8	0,003
0,5	0,964	1,2	0,112	1,9	0,002
0,6	0,864	1,3	0,068	2,0	0,001

Распределение χ^2 дает возможность оценить степень согласованности теоретического и статистического распределений. Будем исходить из того, что величина X действительно распределена по закону $F(x)$. Тогда вероятность p , определенная по таблице, есть вероятность того, что за счет чисто случайных причин мера расхождения теоретического и статистического распределений будет не меньше, чем фактически наблюдаемое в данной серии опытов значение χ^2 . Если эта вероятность p весьма мала (настолько мала, что событие с такой вероятностью можно считать практически невозможным), то результат опыта следует считать противоречащим гипотезе H о том, что закон распределения величины X есть $F(x)$. Эту гипотезу следует отбросить как неправдоподобную. Напротив, если вероятность p сравнительно велика, можно признать расхождения между теоретическим и статистическим распределениями несущественными и отнести их за счет случайных причин. Гипотезу H о том, что величина X распределена по закону $F(x)$, можно считать правдоподобной или, по крайней мере, не противоречащей опытным данным.

Таким образом, схема применения критерия χ^2 к оценке согласованности теоретического и статистического распределений сводится к следующему:

1) определяется мера расхождения χ^2 по формуле, приведенной ранее;

2) определяется число степеней свободы r как число разрядов k минус число наложенных связей s :

$$r = k - s ;$$

3) по r и χ^2 с помощью табл. 3.5 определяется вероятность того, что величина, имеющая распределение χ^2 с r степенями свободы, превзойдет данное значение χ^2 (если эта вероятность весьма мала, гипотеза отбрасывается как неправдоподобная; если эта вероятность относительно велика, гипотезу можно признать не противоречащей опытным данным).

Насколько мала должна быть вероятность p для того, чтобы отбросить или пересмотреть гипотезу, – вопрос неопределенный; он не может быть решен из математических соображений, так же как и вопрос о том, насколько мала должна быть вероятность события для того, чтобы считать его практически невозможным. На практике, если p оказывается меньшим чем 0,1, рекомендуется проверить эксперимент, если возможно – повторить его и в случае, если заметные расхождения снова появятся, пытаться искать более подходящий для описания статистических данных закон распределения.

Следует особо отметить, что с помощью критерия χ^2 (или любого другого критерия согласия) можно только в некоторых случаях опровергнуть выбранную гипотезу H и отбросить ее как явно несогласную с опытными данными; если же вероятность p велика, то этот факт сам по себе ни в коем случае не может считаться доказательством справедливости гипотезы H , а указывает только на то, что гипотеза не противоречит опытным данным.

С первого взгляда может показаться, что чем больше вероятность p , тем лучше согласованность теоретического и статистического распределений и тем более обоснованным следует считать выбор функции $F(x)$ в качестве закона распределения случайной величины. В действительности это не так. Допустим, например, что, оценивая согласие теоретического и статистического распределений по критерию χ^2 , получили $p = 0,99$. Это значит, что с вероятностью 0,99 за счет чисто случайных причин при данном числе опытов должны были получиться расхождения большие, чем наблюдаемые. Мы же получили относительно весьма малые расхождения, которые слишком малы для того, чтобы признать их правдоподобными. Разумнее признать, что столь близкое совпадение теоретического и статистического распределений не является случайным и может быть объяснено определенными причинами, связанными с регистрацией и обработкой опытных данных (в частности, с весьма распространенной на практике «подчисткой» опыт-

ных данных, когда некоторые результаты произвольно отбрасываются или несколько изменяются).

Разумеется, все эти соображения применимы только в тех случаях, когда количество опытов n достаточно велико (порядка нескольких сотен) и когда имеет смысл применять сам критерий, основанный на предельном распределении меры расхождения при $n \rightarrow \infty$. Заметим, что при пользовании критерием χ^2 достаточно большим должно быть не только общее число опытов n , но и числа наблюдений m_i в отдельных разрядах. На практике рекомендуется иметь в каждом разряде не менее 5–10 наблюдений. Если числа наблюдений в отдельных разрядах очень малы (порядка 1–2), имеет смысл объединить некоторые разряды.

Кроме критерия χ^2 , для оценки степени согласованности теоретического и статистического распределений на практике применяется еще ряд других критериев. Из них мы вкратце остановимся на критерии А.Н. Колмогорова.

В качестве меры расхождения между теоретическим и статистическим распределениями А.Н. Колмогоров рассматривает максимальное значение модуля разности между статистической функцией распределения $F^*(x)$ и соответствующей теоретической функцией распределения:

$$D = \max |F^*(x) - F(x)|.$$

Основанием для выбора в качестве меры расхождения величины D является простота ее вычисления. Вместе с тем она имеет достаточно простой закон распределения. А.Н. Колмогоров доказал, что какова бы ни была функция распределения $F(x)$ непрерывной случайной величины X , при неограниченном возрастании числа независимых наблюдений n вероятность неравенства

$$D\sqrt{n} \geq \lambda$$

стремится к пределу

$$p(\lambda) = 1 - \sum_{k=-\infty}^{\infty} (-1)^k e^{-2k^2\lambda^2}.$$

Значение вероятности $p(\lambda)$ приведены в табл. 3.4.

Схема применения критерия А.Н. Колмогорова следующая: строится статистическая функция распределения $F^*(x)$ и предпо-

лагаемая теоретическая функция распределения $F(x)$, и определяется максимум D модуля разности между ними (рис. 3.7).

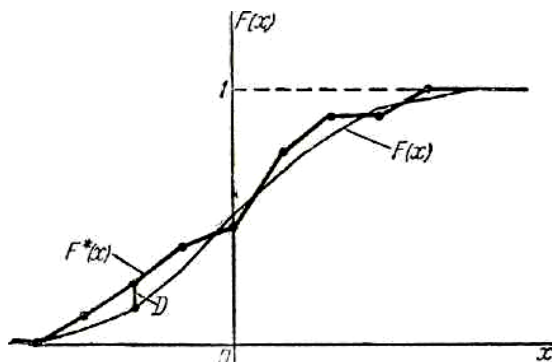


Рис. 3.7. Теоретическая и статистическая функция распределения

Далее, определяется величина

$$\lambda = D\sqrt{n}$$

и по табл. 3.4 находится вероятность $P(X)$. Это есть вероятность того, что (если величина X действительно распределена по закону $F(x)$) за счет чисто случайных причин максимальное расхождение между $F^*(x)$ и $F(x)$ будет не меньше, чем фактически наблюдаемое. Если вероятность $P(X)$ весьма мала, гипотезу следует отвергнуть как неправдоподобную; при сравнительно больших $P(\lambda)$ ее можно считать совместимой с опытными данными.

Критерий А.Н. Колмогорова своей простотой выгодно отличается от описанного ранее критерия χ^2 ; поэтому его весьма охотно применяют на практике. Следует, однако, сказать, что этот критерий можно применять только в случае, когда гипотетическое распределение $F(x)$ полностью известно заранее из каких-либо теоретических соображений, т.е. когда известен не только вид функции распределения $F(x)$, но и все входящие в нее параметры. Такой случай сравнительно редко встречается на практике. Обычно из теоретических соображений известен только общий вид функции $F(x)$, а входящие в нее числовые параметры определяются по данному статистическому материалу. При применении критерия χ^2 это обстоятельство учитывается соответствующим уменьшением числа

Таблица 3.5

p c	0,99	0,98	0,95	0,90	0,80	0,70	0,50	0,30	0,20	0,10	0,05	0,02	0,01	0,001
1	0,	0,	0,	0,	0,	0,	0,	1,074	1,642	2,71	3,84	5,41	6,64	10,83
2	0,	0,	0,	0,	0,	0,	1,386	2,41	3,22	4,60	5,99	7,82	9,21	13,82
3	0,	0,	0,	0,	1,005	1,424	2,37	3,66	4,64	6,25	7,82	9,84	11,34	16,27
4	0,	0,	0,	1,064	1,649	2,20	3,36	4,88	5,99	7,78	9,49	11,67	13,28	18,46
5	0,	0,	1,145	1,610	2,34	3,00	4,35	6,06	7,29	9,24	11,07	13,39	15,09	20,05
6	0,	1,134	1,635	2,20	3,07	3,83	5,35	7,23	8,56	10,64	12,59	15,03	16,81	22,5
7	1,239	1,564	2,14	2,83	3,82	4,67	6,35	8,38	9,80	12,02	14,07	16,62	18,48	24,3
8	1,646	2,03	2,73	3,49	4,59	5,53	7,34	9,52	11,03	13,36	15,51	18,17	20,1	26,1
9	2,09	2,53	3,32	4,17	5,38	6,39	8,34	10,66	12,24	14,68	16,92	19,68	21,7	27,9
10	2,56	3,06	3,94	4,86	6,18	7,27	9,34	11,78	13,44	15,99	18,31	21,2	23,2	29,6
11	3,05	3,61	4,58	5,58	6,99	8,15	10,34	12,90	14,63	17,28	19,68	22,6	24,7	31,3
12	3,57	4,18	5,23	6,30	7,81	9,03	11,34	14,01	15,81	18,55	21,0	24,1	26,2	32,9
13	4,11	4,76	5,89	7,04	8,63	9,93	12,34	15,12	16,98	19,81	22,4	25,5	27,7	34,6
14	4,66	5,37	6,57	7,79	9,47	10,82	13,34	16,22	18,15	21,1	23,7	26,9	29,1	36,1
15	5,23	5,98	7,26	8,55	10,31	11,72	14,34	17,32	19,31	22,3	25,0	28,3	30,6	37,7
16	5,81	6,61	7,96	9,31	11,15	12,62	15,34	18,42	20,5	23,5	26,3	29,3	32,0	39,3
17	6,41	7,26	8,67	10,08	12,00	13,53	16,34	19,51	21,6	24,8	27,6	31,0	33,4	40,8
18	7,02	7,91	9,39	10,86	12,86	14,44	17,34	20,6	22,8	26,0	28,9	32,3	34,8	42,3
19	7,63	8,57	10,11	11,65	13,72	15,35	18,34	21,7	23,9	27,2	30,1	33,7	36,2	43,8
20	8,26	9,24	10,85	12,44	14,58	16,27	19,34	22,8	25,0	28,4	31,4	35,0	37,6	45,3
21	8,90	9,92	11,59	13,24	15,44	17,18	20,3	23,9	26,2	29,6	32,7	36,3	38,9	46,8
22	9,54	10,60	12,34	14,04	16,31	18,10	21,3	24,9	27,3	30,8	33,9	37,7	40,3	48,3
23	10,20	11,29	13,09	14,85	17,19	19,02	22,3	26,0	28,4	32,0	35,2	39,0	41,6	49,7
24	10,86	11,99	13,85	15,66	18,06	19,94	23,3	27,1	29,6	33,2	36,4	40,3	43,0	51,2
25	11,52	12,70	14,61	16,47	18,94	20,9	24,3	28,2	30,7	34,4	37,7	41,7	44,3	52,6
26	12,20	13,41	15,38	17,29	19,82	21,8	25,3	29,2	31,8	35,6	38,9	42,9	45,6	54,1
27	12,88	14,12	16,15	18,11	20,7	22,7	26,3	30,3	32,9	36,7	40,1	44,1	47,0	55,5
28	13,56	14,85	16,93	18,94	21,6	23,6	27,3	31,4	34,0	37,9	41,3	45,4	48,3	56,9
29	14,26	15,57	17,71	19,77	22,5	24,6	28,3	32,5	35,1	39,1	42,6	46,7	49,6	58,3
30	14,95	16,31	17,49	20,6	23,4	25,5	29,3	33,5	36,2	40,3	43,8	48,0	50,9	59,7

степеней свободы распределения χ^2 . Критерий А.Н. Колмогорова такого согласования не предусматривает. Если все же применять этот критерий в тех случаях, когда параметры теоретического распределения выбираются по статистическим данным, критерий дает заведомо завышенные значения вероятности $P(\lambda)$; поэтому мы в ряде случаев рискуем принять как правдоподобную гипотезу, в действительности плохо согласующуюся с опытными данными.

В табл. 3.5 приведены значения χ^2 в зависимости от r и p .

Контрольные вопросы

1. Что такое моделирование и какие разновидности модели знаете?
2. Дайте общее описание математической модели, структурных и функциональных моделей.
3. Дайте краткую характеристику математических моделей.
4. Перечислите основные методы многовариантного анализа и кратко их охарактеризуйте.
5. Расскажите о методе статистических испытаний Монте-Карло – основных направлениях применения.
6. Опишите обобщенную структуру алгоритма метода Монте-Карло.
7. Как получить последовательности случайных чисел с различными законами распределения?
8. Опишите методы получения случайных чисел, равномерно распределенных на отрезке $[0, 1]$.
9. Дайте определения понятий статистического ряда и гистограммы.
10. Какие знаете методы проверки качества случайных чисел, каково общее представление о критериях согласия?

ГЛАВА 4. МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САПР – РЕШЕНИЕ ЗАДАЧИ СИНТЕЗА, МЕТОДЫ ОПТИМИЗАЦИИ

4.1. Введение в проблему параметрического синтеза

4.1.1. Общие сведения, постановка задачи

Синтез системы – выбор определенного варианта (принятие решения) – основная проблема проектирования. Применение математических методов при решении этой задачи находит все более широкое применение в различных областях науки и техники. Это сравнительно молодая область математики, ориентированная на компьютерное моделирование, требующая высокопроизводительных средств вычислительной техники. Наибольший успех и широкое внедрение в практику получило решение этих задач в экономике (распределение ресурсов, транспортные задачи и т.д.). Хотя математические модели в этой области сравнительно просты, процедуры поиска оптимальных решений и их определение, отнюдь, не тривиальны и потребовали оригинальных подходов. Именно в связи с решением подобных задач известный советский математик, академик Л.В. Канторович был удостоен Нобелевской премии по экономике.

Применением математических методов для обоснования решений во всех областях человеческой деятельности занимается раздел научных методов под общим названием «Исследование операций». Это междисциплинарное направление, возникновению которого предшествовало становление теории систем. Предметом исследования является разработка методов анализа целенаправленных действий (операций) и объективная сравнительная оценка полученных решений. Распространение идей исследования операций совпало с развитием методов математического программирования, которое в отличие от классических математических методов, включает неко-

торые средства постановки задачи, и позволяет получать область допустимых решений и различные варианты этих решений.

Особенности исследования операций заключаются в следующем:

- предполагается разработка нескольких вариантов решений, несколько путей достижения целей;
- при выборе решения допускается учет не только количественных, но и качественных критериев, что позволяет обеспечивать большее соответствие решаемой задачи реальной действительности и большую объективность.

Основным математическим аппаратом исследования операций является математическое программирование. Это раздел теории оптимизации (теории экстремальных задач), занимающийся изучением и решением задач минимизации (максимизации) функций нескольких переменных на подмножестве векторного пространства, заданного в виде системы уравнений или системы неравенств.

Цель всех методов – определить оптимальное решение, т.е. решение, которое по тем или иным признакам предпочтительнее перед другими. Параметры, совокупность которых образуют решения, называют *элементами решения*.

Для представления задачи в символическом виде обозначим множество возможных решений с помощью вектора X , каждую отдельную совокупность решений – через x . Для сравнения между собой различных решений вводится количественный критерий показателя эффективности $f(x)$ или целевая функция. Возможны два варианта формирования целевой функции – поиск ее наибольшего или наименьшего значения. На практике, в основном, используется второй вариант, при этом нахождение наименьшего значения целевой функции означает, что найденные значения параметров системы наилучшим образом приближаются к желаемым значениям параметров системы (в целевую функцию входят разности между желаемыми и реализованными значениями параметров системы).

Как правило, показатель эффективности зависит от двух факторов:

$$f = f(\alpha, x), \quad (4.1)$$

α – заданные заранее известные факторы (детерминированные или стохастические); x – одна из совокупностей проектных решений.

Детерминированными называются задачи, когда вся исходная информации полностью определена; *стохастическими* – задачи, когда исходная информация содержит элементы неопределенности или некоторые параметры носят вероятностный характер.

Задача выбора совокупности решений формируется следующим образом. При заданном комплексе условий α найти такое решение $x = x^*$, при котором показатель эффективности $f(\alpha, x)$ принимает минимальное значение

$$f(\alpha, x^*) = \min(f(\alpha, x)), \quad x \in X. \quad (4.2)$$

Точку x^* называют *точкой локального экстремума*, а функцию $f(\alpha, x)$ – *одноэкстремальной (унимодальной)*, если она имеет один экстремум, и *многоэкстремальной*, если она имеет более одного минимума (максимума). Соответственно, выделяют одно- и многоэкстремальные задачи оптимизации.

В общем виде это классическая математическая задача нахождения минимума функции или функционала принадлежит к классу так называемых классических математических вариационных задач, давно разрабатываемых математиками. Числовая функция, определенная на множестве функций, называется *функционалом*. Чтобы найти экстремум функции, необходимо продифференцировать модель по всем переменным, приравнять каждое уравнение к нулю и решить полученную систему. Пример из области ускорителей заряженных частиц – выбор функции, описывающий изменение амплитуды напряженности ускоряющего поля и фазовой скорости электромагнитной волны по длине группирователя (диафрагмированного волновода) с целью обеспечения минимального фазового размера и энергетического разброса сгустка ускоряемых заряженных частиц и максимального набора энергии пучком, которые являются функциями геометрических размеров ускоряющей структуры и амплитуды ускоряющего поля. Однако использование классических методов вариационного исчисления в ряде случаев на практике затруднено, а иногда и невозможно по ряду причин:

- когда параметров много, задача решения системы напрямую может оказаться непростой;
- когда на элементы решения наложены ограничения (оптимум (наименьшее значение целевой функции) может находиться на гра-

нице области поиска, т.е. целевая функция может вообще не иметь экстремума, при этом возникает многомерная вариационная задача при ограничениях, трудно реализуемая традиционными методами);

- в некоторых задачах функция $f(\alpha, x)$ вообще не имеет производных, например для целочисленного значения аргумента.

Все эти причины делают задачу поиска, отнюдь, не тривиальной, и классические методы вариационного исчисления оказываются малоэффективными. Это и привело к появлению альтернативных, чисто машинных (вычислительных) методов математического программирования – методов, основанных на использовании модели, как средства анализа и специальных процедур выбора параметров, как средства автоматизированного поиска решения (самый примитивный способ – сканирование). Это не строгие, а так называемые эвристические методы, основанные на здравом смысле, интуиции и аналогиях.

В математическом программировании сложились понятия следующих разделов.

Линейное программирование – целевая функция линейна, а множество, на котором ищется экстремум целевой функции, задается системой линейных равенств и неравенств. В связи с широким использованием линейного программирования сложились целые классы задач, структура которых позволяет создать специальные методы их решения, в отличие от методов решения задач общего характера. Линейное программирование используется для решения различных военных, промышленных, экономических и организационных задач. Например, такими задачами являются:

- транспортная задача – план перевозки однородного груза, чтобы стоимость перевозок была минимальной;
- задача оптимального планирования производства – сколько единиц каждого вида продукции должно произвести предприятие, чтобы прибыль была максимальной;
- задача о размещении – согласование размещения пунктов потребления продукции с транспортной задачей и задачей оптимального планирования производства;
- задача о посевной площади – требуется определить, какую площадь на каждом участке нужно отвести под ту или иную куль-

туру, чтобы прибыль была максимальна, а урожай по каждому виду культуры был не меньше, чем установлено по плану.

Нелинейное программирование – целевая функция и ограничения нелинейны. Нелинейное программирование принято подразделять следующим образом:

- *выпуклое программирование* – выпукла целевая функция и выпукло множество, на котором решается задача минимизации;
- *квадратичное программирование* – целевая функция квадратична, а ограничения задаются в виде линейных равенств и неравенств (особый класс задач выпуклого программирования, часто встречающийся на практике);
- *многоэкстремальные задачи* – различного вида специализированные задачи, например задачи минимизации на выпуклом множестве вогнутых функций.

Целочисленное программирование – особый вид математического программирования, когда на переменные накладываются условия целочисленности.

Динамическое программирование – решение многоэтапных задач, когда неоптимальность на каждом этапе в отдельности может привести к оптимальности в целом.

Можно выделить два типа задач оптимизации – задачи безусловной оптимизации (без ограничений) и задачи условной оптимизации, когда на параметры наложены ограничения в виде различных типов равенств и неравенств. Ограничения объективно появляются при проектировании технических объектов и вытекают из конкретной физической и технологической реализуемости внутренних параметров элементов, ограниченности ресурсов и т.п. При постановке задачи оптимизации учет ограничений иногда бывает принципиально необходим. Так, если целевая функция имеет вид $f(x) = a + bx$ и не наложены ограничения на параметр x , то задача поиска экстремального значения $f(x)$ становится некорректной. Ограничения суживают область поиска оптимального решения, и искомый экстремум становится условным. Различают прямые и функциональные ограничения.

Прямые ограничения имеют вид

$$x_{hi} \leq x_i \leq x_{bi} \quad \text{при} \quad i \in [1 \div n], \quad (4.3)$$

где x_{hi}, x_{bi} – минимально и максимально допустимые значения i -го управляемого параметра; n – размерность пространства варьируемых параметров. Например, для многих объектов параметры элементов не могут быть отрицательными: $x_{hi} > 0$ (геометрические размеры, электрические сопротивления, массы и т.п.).

Функциональные ограничения, как правило, представляют собой условия работоспособности выходных параметров, не вошедших в целевую функцию. Функциональные ограничения могут быть типа:

равенств

$$\varphi(x) = 0 ; \quad (4.4)$$

неравенств

$$\varphi(x) > 0 . \quad (4.5)$$

В подавляющем большинстве случаев задачи параметрической оптимизации технических объектов сводятся к задачам условной оптимизации. Решение задач можно выполнить с помощью одного из двух подходов. В первом подходе учитывается, что большинство развитых методов оптимизации ориентировано на поиск безусловного экстремума, поэтому их применение к решению задачи условной оптимизации требует, чтобы эта задача была предварительно сведена к задаче безусловной оптимизации. Во втором подходе используются методы, специально разработанные для решения задач нелинейного программирования с ограничениями.

Методы решения весьма разнообразны по стратегии поиска минимума целевой функции, и в этом смысле можно выделить методы общего характера и методы, адаптированные к конкретным задачам. Большое количество работ посвящено исследованию методов оптимизации [1–12] – вопросам сходимости, быстродействия и устойчивости. Значительный класс задач связан с применением стохастических методов поиска, а также с применением генетических алгоритмов для решения технических задач. Особый класс представляет решение задач, в которых целевая функция имеет разрывы. В дальнейшем изложении материала на основе литературных данных приведем краткий обзор основных методов оптимизации, применяемых в науке и технике, уделяя основное внимание нелинейному программированию.

4.1.2. Выбор целевой функции

Существует целый ряд методов, позволяющих для специального класса задач, получить решение с определенной степенью точности. Однако, по-видимому, невозможно создать единый метод, позволяющий получить точное решение общей задачи нелинейного программирования за конечное число шагов.

Одна из основных проблем постановки экстремальных задач заключается в формулировке целевой функции. Сложность выбора целевой функции состоит в том, что любой технический объект первоначально имеет векторный характер критериев оптимальности (многокритериальность). Причем улучшение одного из выходных параметров, как правило, приводит к ухудшению другого, так как все выходные параметры являются функциями одних и тех же варьируемых параметров и не могут изменяться независимо друг от друга. Такие выходные параметры называют *конфликтными*. Сведение многокритериальной задачи к однокритериальной называют *сверткой векторного критерия*. Задача поиска его экстремума сводится к задаче математического программирования.

В зависимости от того, каким образом выбираются и объединяются выходные параметры в скалярной функции качества, различают частные, аддитивные, мультипликативные, минимаксные, статистические критерии и т.д. Объединение нескольких выходных параметров, имеющих в общем случае различную физическую размерность, в одной скалярной целевой функции требует предварительного нормирования этих параметров. Наиболее распространены из них следующие критерии.

Взвешенный аддитивный критерий применяют, когда можно выделить две группы выходных параметров. Полагаем, что все $y_j(x)$ безразмерны. Тогда для случая минимизации целевой функции свертка векторного критерия будет иметь вид

$$f(x) = \sum_{j=1}^q a_j y_j^-(x) + \sum_{j=q+1}^m a_j y_j^+(x), \quad (4.6)$$

где $a_j > 0$ – весовой коэффициент, определяющий степень важности j -го параметра (обычно a_j выбираются проектировщиком и в процессе остаются постоянными).

В первую группу входят выходные параметры, значения которых в процессе оптимизации нужно увеличивать $y_j^+(x)$, например производительность, помехоустойчивость, вероятность безотказной работы и т.п. Во вторую – выходные параметры, значения которых следует уменьшать $y_j^-(x)$, например расход топлива, длительность переходного процесса, перерегулирование, смещение и пр.

Мультипликативный критерий может применяться в тех случаях, когда отсутствуют условия работоспособности типа равенств, и выходные параметры не могут принимать нулевые значения. Тогда минимизируемая мультипликативная целевая функция имеет вид

$$f(x) = \prod_{j=1}^q y_j^-(x) / \prod_{j=q+1}^m y_j^+(x). \quad (4.7)$$

Большую ценность для разработчика представляют результаты оптимизации по статистическим критериям, в которых целевой функцией является вероятность выполнения всех заданных условий работоспособности. Однако использование этих критериев в процессе оптимизации связано с проведением многократного статистического анализа методом Монте-Карло, что для достаточно сложных моделей анализа требует использования компьютеров очень высокой производительности.

Какие вычислительные сложности возникают при разработке алгоритма расчета?

Во-первых – размерность задачи. Можно считать, что задача с числом переменных менее 15 является малой задачей. При числе переменных в диапазоне от 15 до 50 – средняя задача, и при числе переменных более 50 – большая задача. Эта градация весьма условна, так как даже при числе переменных равным двум найти решение может оказаться достаточно сложно.

Во-вторых – вычисление производных целевой функции (в тех алгоритмах, где они используются). Во многих задачах целевая функция формируется на основе численного моделирования на ЭВМ. Например, при расчете волноводного группирователя линей-

ного ускорителя электронов (ЛУЭ) целевая функция базируется на расчетных параметрах сгустков электронов, которые определяются из решения (в самой простой модели сгустков) системы обыкновенных дифференциальных уравнений.

В-третьих – решение многих нелинейных задач, как правило, довольно «дорогостоящая» (большое время счета) процедура из-за многократного вычисления целевой функции или из-за того, что сама задача заключается в решении многих подзадач, связанных между собой. При расчете линейного ускорителя электронов, например, это связано с необходимостью выбора изменения амплитуды высокочастотного ускоряющего поля и фазовой скорости волны по длине ускоряющей структуры. Выбор этих характеристик базируется на табличных экспериментальных данных.

В-четвертых – возможное плохое масштабирование задач, в которых диапазоны изменений отдельных переменных отличаются друг от друга на несколько порядков. Например, одна переменная изменяется в диапазоне от 1 до 10, вторая – от 10^6 до 10^7 . *Масштабирование* – преобразование переменных с помощью соответствующих коэффициентов к такому виду, когда значения переменных оказываются сопоставимы по своей величине, а значение целевой функции измеряется в безразмерных величинах, близких к единице.

Решение любой задачи состоит из нескольких этапов и начинается с корректной постановки инженерной задачи. Необходимо четко представлять преимущества, недостатки и специфические особенности различных методов оптимизации. Каждой задаче должен соответствовать адекватный метод решения.

Выбор целевой функции непосредственно связан и вытекает из постановки решаемой задачи, начиная с самого первого ее этапа – построения упрощенной модели решаемой задачи, отражающей основные действующие факторы, и установление закономерностей, которыми определяется поведение рассматриваемой системы. В этом вопросе нужно исходить из принципа перехода от простого к сложному. Чтобы сузить область поисков минимума, необходимо построить ряд дополняющих и уточняющих моделей, т.е. каждая последующая модель отражает все свойства предшествующей модели и отличается от нее включением в анализ более тонких и сложных параметров реальной системы. Это позволяет значительно со-

кратить область первоначального поиска минимума. Вообще говоря, любая модель является упрощенным образом реальной системы, поэтому необходимо оценить факторы, которые можно не учитывать как слабо влияющие на работу системы в целом. Построение модели начинается с определения границ изучаемой системы. Границы служат средством выделения системы из окружающей среды. Между системой и внешней средой всегда существуют взаимосвязи. Определение границ является первым приближением описания реальной системы. Взаимосвязь между системой и внешней средой, таким образом, фиксируется на определенном уровне. Конкретный выбор границ осуществляется, в самом общем случае, методом последовательных приближений на основе анализа численных экспериментов. В процессе расчетов может оказаться, что границы выбраны слишком жесткими. В этом случае нужно произвести коррекцию первоначального варианта. Таким образом, оптимизационное исследование приобретает итерационный характер.

Следующим этапом является математическая запись выбранной модели – установление соотношений между параметрами модели. Например, в приложении к физическим задачам математическими моделями часто являются системы дифференциальных и линейных алгебраических уравнений.

На основе предшествующих этапов производится построение целевой функции, числовое значение которой (минимум или максимум) соответствует оптимальному варианту решения задачи. Это основной показатель оптимизационного исследования, позволяющий выбрать компромиссный вариант из возможных решений задачи. К любой системе могут предъявляться взаимоисключающие требования. Целевая функция состоит, как правило, из нескольких параметров, значения которых должны стремиться к достижению требуемых значений (параметров цели). Степень приближения полученного решения к параметрам цели зависит от формы построения целевой функции. Численное значение целевой функции является единственным критерием, который может использоваться при определении оптимального решения задачи. Возможны различные варианты формирования целевой функции – во-первых, все параметры входят в построение целевой функции, во-вторых, определяется степень важности каждого параметра, и часть из них выводится в раздел ограничений. Например, один из параметров системы

выбирается как основной и по нему ведется оптимизация, а остальные рассматриваются как ограничения. Другой путь – формирование обобщенной целевой функции, содержащей несколько параметров цели, которая учитывает конкурирующие требования при их достижении. При этом необходимо продумать механизм их совместимости. Это напрямую связано с выбором независимых переменных, которые позволяют адекватно описать поведение выбранной системы. Переменные могут быть двух типов – переменные, значения которых могут изменяться в достаточно широком диапазоне, и переменные, значения которых фиксированы и определяются внешними факторами. Часть переменных может иметь вероятностный характер. Оценивая степень важности каждого параметра системы, необходимо выбирать независимые переменные таким образом, чтобы незначительные их значения не перегружали задачу. При выборе целевой функции и независимых переменных строится модель, описывающая связь между переменными задачи и влияние независимых переменных на степень достижения конечной цели.

Часто целевая функция имеет экономическую составляющую. Например, требование высокого качества произведенного товара (максимальность) всегда конкурирует с минимизацией материальных расходов и энергопотребления. В задачах создания уникальных физических установок или устройств достижение необходимых технических параметров может являться значительно более весомым фактором, чем материальные и экономические затраты.

Обычно в физических задачах приходится сравнивать различные по своей природе параметры, например внешние габариты устройства или установки и ее энергетические характеристики. Возможный путь решения – обезразмеривание параметров системы за счет введения нормирующих коэффициентов и весовых коэффициентов.

Дальнейшим этапом является численный эксперимент и оценка результатов расчетов, которая устанавливает практическую достоверность как выбранной модели, так и целевой функции. Последующая стратегия поиска заключается либо в принятии выбранной модели и целевой функции, либо в их уточнении, которое связано с усложнением модели и целевой функции за счет ранее не учитываемых свойств рассматриваемой системы. Несмотря на универсальность методов оптимизации, их успешное применение к реше-

нию конкретной задачи зависит от глубины понимания физической сути задачи и четкого представления об области применения того или иного алгоритма поиска оптимума.

4.1.3. Задачи нелинейного программирования

Задачам нелинейного программирования в настоящее время уделяется значительное внимание. Предложен ряд методов, позволяющих с определенной точностью получать решения для специального класса задач. Надежные и одновременно экономичные методы поиска глобального экстремума в настоящее время неизвестны. Практически большинство используемых методов является методами локального поиска (выпуклого программирования).

Задачи поиска оптимальных параметров различных устройств и систем связаны с анализом функций, определенных на множествах конечномерного евклидова пространства. В дальнейшем рассмотрении материала основное внимание будет уделено нелинейному программированию и особенно вопросам квадратичного программирования, так как многие задачи могут быть сведены к квадратичному программированию. Определим основные понятия выпуклого анализа, которые используются в дальнейшем.

Наборы $x^T = (x_1, x_2, \dots, x_n)$ называются *точками (векторами)*, а числа $x_1, x_2, \dots, x_n$ — их *координатами*.

Совокупность всех наборов $x^T = (x_1, x_2, \dots, x_n)$ n вещественных чисел $x_1, x_2, \dots, x_n$ называют *евклидовым пространством размерности n* , если выполняются следующие условия.

Пусть $x \in E_n$, $y \in E_n$ и α — вещественное число. Тогда:

$$x + y = (x_1 + y_1, x_2 + y_2, \dots, x_n + y_n) \text{ (сложение);}$$

$$\alpha x = (\alpha x_1, \alpha x_2, \dots, \alpha x_n) \text{ (умножение на число);}$$

$$\langle x, y \rangle = \sum_{i=1}^n x_i \cdot y_i \text{ (скалярное произведение).}$$

Множество X n -мерного евклидова пространства E_n называется *выпуклым*, если с любыми двумя точками $x \in X$ и $y \in X$ ему принадлежит и соединяющий их отрезок $[x, y]$.

Выпуклость множества X означает, что из $x, y \in X$ следует $z = \alpha x + (1 - \alpha)y \in X$ для всех $0 \leq \alpha \leq 1$.

В E_2 , например, выпуклы отрезок, полупрямая, прямая, круг, треугольник, полуплоскость и вся плоскость.

Функция $f(x)$, определенная на выпуклом множестве X , называется *выпуклой*, когда для любых $x, y \in X$ и всех $\alpha \in [0, 1]$ выполняется неравенство

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y). \quad (4.8)$$

Очевидно, что каждая точка любой хорды графика функции $f(x)$ лежит либо над графиком, либо принадлежит ему (рис. 4.1). Если для любого $\alpha \in (0, 1)$ неравенство (4.8) – строгое, то функция $f(x)$ – *строго выпуклая*. Примером выпуклой функции является квадратичная функция.

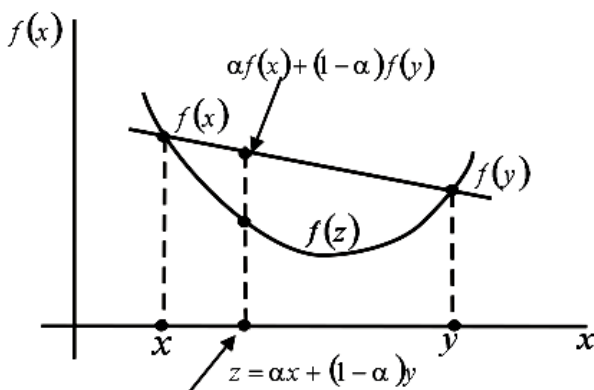


Рис. 4.1. Выпуклая функция

Вогнутой функцией называется такая функция $f(x)$, для которой $-f(x)$ выпукла. Следовательно, если

$$f(\alpha x + (1 - \alpha)y) \geq \alpha f(x) + (1 - \alpha)f(y) \quad (4.9)$$

для любых $x, y \in X$ и всех $\alpha \in [0, 1]$, то $f(x)$ – *вогнутая* функция.

Основная задача математического программирования заключается в следующем.

Дано множество $X = \{x : \varphi_i(x) \geq 0, i = \overline{1, m}\}$, где $\varphi_i(x)$, $i = \overline{1, m}$ – заданные скалярные функции.

Пусть скалярная функция $f(x)$ определена на множестве X .

Минимизация функции $f(x)$ на множестве X называется *основной задачей* математического программирования. Существуют различные формы записи этой задачи:

$$f(x) \rightarrow \min, x \in X, \quad (4.10)$$

или эквивалентная вторая форма записи

$$\min \{f(x) : x \in X\} \quad \text{и} \quad \min_{x \in X} f(x). \quad (4.11)$$

Эти формы записи означают, что ставится следующая задача.

1. Либо найти точку минимума функции $x^* \in X$

$$f(x^*) = \min_{x \in X} f(x),$$

x^* является точкой минимума функции $f(x)$ на множестве X , если $f(x^*) \leq f(x)$ для всех $x \in X$. Таких точек может быть несколько и даже бесконечное множество (в зависимости от свойств множества X и функции). Множество всех минимумов $f(x)$ на X обозначим как X^* .

2. Либо если не существует такая точка x^* , то найти

$$f^* = \inf_{x \in X} f(x).$$

Это означает, что $f(x)$ не ограничена снизу на множестве X и в качестве нижней грани принимается значение $(-\infty)$.

3. Либо убедиться, что $X = \emptyset$ (множество пусто).

Пример 1. Пусть функция $f(x) = \sin^2(\pi/x)$ определена на множестве $X = \{x : 1 \leq x \leq 2\}$. Минимальное значение равно нулю, множество X^* состоит из единственной точки $x^* = 1$.

Пример 2. Пусть функция $f(x) = \ln(x)$ определена на множестве $X = \{x : 0 < x \leq 1\}$. В этом случае $X^* = \emptyset$, так как во всех точках из X функция принимает конечные значения, а для последовательности $x_k = 1/k$ ($k = 1, 2, \dots$) имеет $\lim_{k \rightarrow \infty} (x_k) = -\infty$.

Пример 3. Функция $f(x) = |x| + |x - 1| - 1$, определенная на множестве $X = \{x : |x| \leq 1\}$ принимает свое наименьшее значение, равное нулю, во всех точках отрезка $X^* = \{x : 0 \leq x \leq 1\}$.

Если $X = \{x : 1 \leq x \leq 2\}$, то X^* содержит только одну точку $x^* = 1$. Если $X = \{x : 1 < x \leq 2\}$, то $X^* = \emptyset$.

В настоящее время существует множество разнообразных алгоритмов нахождения минимумов одно- и многомерных функций [10–21]. Выбор того или иного алгоритма для решения задачи является сам по себе достаточно сложной проблемой, так как необходимо адекватно сопоставить физические свойства модели с особенностями и возможностями выбранного метода поиска минимума функции. Далее будут рассмотрены некоторые наиболее характерные алгоритмы.

Задачи оптимизации, как правило, связаны с нахождением минимума многомерной целевой функции. Весь алгоритм поиска минимума в большинстве случаев можно разделить на отдельные последовательные этапы:

- 1) стратегия поиска минимума многомерной целевой функции (с возможностью нахождения многомерного глобального минимума);
- 2) стратегия поиска минимума одномерной целевой функции (с возможностью нахождения глобального минимума по одной переменной).

В дальнейшем изложении материала рассмотрим (рис. 4.2) сначала алгоритмы поиска минимума одномерной унимодальной целевой функции, затем алгоритмы поиска минимума многомерной целевой функции и в заключение – алгоритмы поиска глобального минимума. На практике реализуются как алгоритмы, не требующие вычисления производных минимизируемой функции, так и алгоритмы с использованием производной. Решить проблему нахождения минимума целевой функции, анализируя выражение первой производной исследуемой функции, не всегда возможно. Это

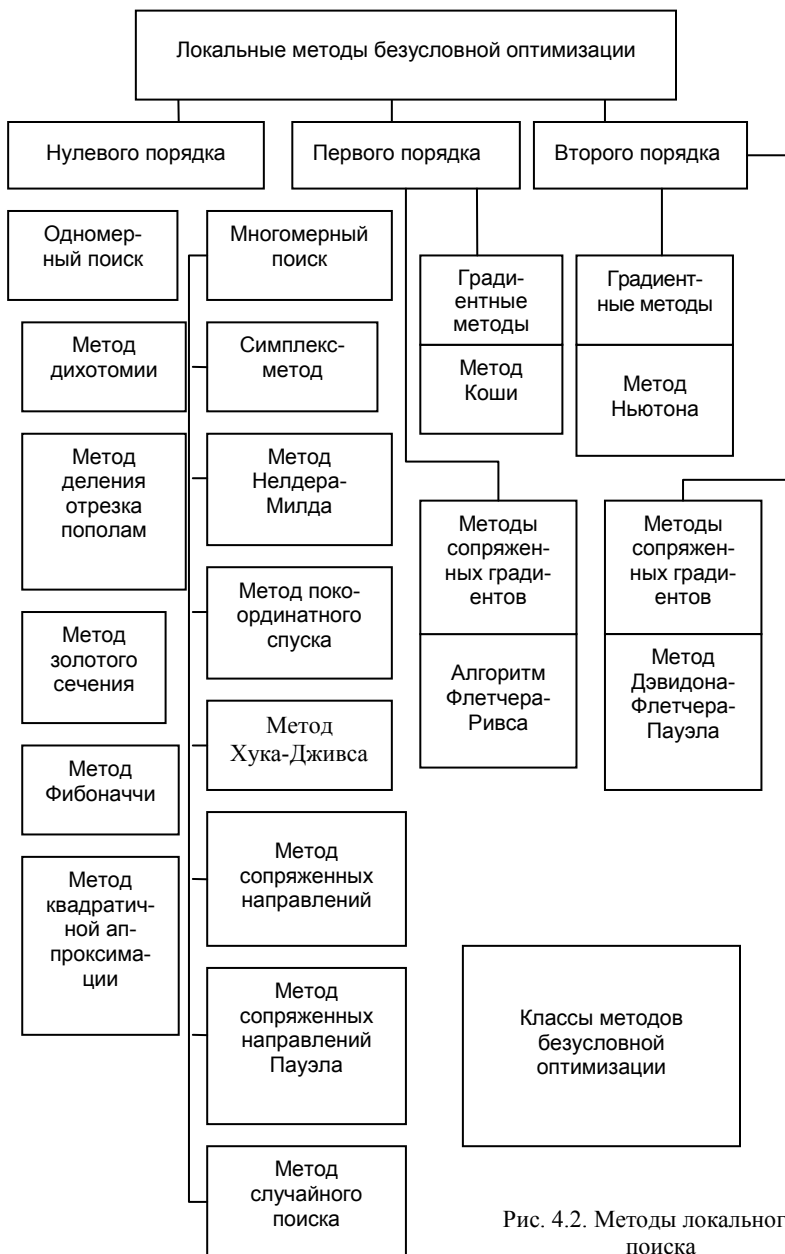


Рис. 4.2. Методы локального поиска

связано с тем, что в реальных практических задачах часто заранее неизвестно является ли рассматриваемая функция дифференцируемой. Кроме того, решение уравнения $f'(x)=0$ может оказаться достаточно сложной задачей. Выбор того или иного алгоритма определяется общим характером решаемой задачи поиска минимума многомерной функции. Поскольку априори в общем случае неизвестно поведение функции на интервале поиска, то необходимо провести предварительное исследование на предмет определения числа возможных минимумов функций и поиска глобального минимума целевой функции.

Алгоритмы решения задачи следует разделять на *принципиальный* и *реализуемый* алгоритмы. Различие между этими понятиями заключается в следующем – на каждой итерации принципиальный алгоритм может осуществлять произвольное число арифметических операций и обращений к функциям, а реализуемый алгоритм только конечное число подобных действий. При этом считается, что приемлемое приближенное значение целевой функции может быть вычислено за приемлемое конечное время.

Процесс конструирования метода расчета заключается в разработке принципиального алгоритма, который затем преобразуется в реализуемый алгоритм таким образом, чтобы свести каждую его итерацию к конечному числу машинных операций.

4.2. Методы поиска минимума одномерных унимодальных функций

4.2.1. Методы исключения интервалов (дихотомия, деление отрезка пополам, золотое сечение, метод Фибоначчи)

Выбор метода зависит от конкретной задачи и выделить априори метод лучший по всем параметрам не представляется возможным. Различаются два подхода в стратегии поиска оптимума:

- 1) определение минимума целевой функции в заданном интервале поиска (с требуемой точностью) на основе анализа ее значений в пробных точках без ограничений их числа (т.е. обращений к вычислению функции);
- 2) определение минимума целевой функции методом, гарантирующим наилучшую точность, при заданном заранее допустимом количестве обращений к вычислению функции.

Рассматриваемые алгоритмы поиска минимума функции одной переменной предполагают наличие только одного минимума функции (унимодальная функция) на исследуемом интервале изменений аргумента. В понятие минимизации функции следует включить не только поиск ее экстремума, но и оценку минимального значения функции на краях исследуемого интервала, т.е. функция может быть монотонно возрастающей или убывающей.

Рассмотрим алгоритмы нахождения локального минимума методом сокращения интервала поиска [10–13]. Все алгоритмы построены на анализе значений минимизируемой функции в пробных точках, расположенных внутри интервала поиска. Различие методов заключается в принципе выбора места расположения этих пробных точек.

Принцип симметричного поиска (симметричного расположения пробных точек) реализуется в задачах оптимизации несколькими способами – методом дихотомии, методом деления отрезка пополам, методом золотого сечения, а также методом Фибоначчи. Алгоритмы различаются быстродействием, которое, в основном, связано с количеством обращений к вычислению минимизируемой функции $f(x)$. В большинстве случаев этот критерий является основным, так как вычисление самой функции составляет, как правило, основное время счета. При этом нужно учитывать, что процесс поиска минимума функции определяется задаваемой точностью расчетов, т.е. конечным размером интервала $[a_{\text{кон}}, b_{\text{кон}}]$.

Дихотомия. Наиболее простой алгоритм, в котором на каждой итерации интервал поиска минимума функции сокращается вдвое. Вычисление значения минимизируемой функции на каждой итерации производится дважды. Суть алгоритма заключается в следующем – на текущей итерации определяется $x_{\text{ср}}$ середина отрезка $[a, b]$ и вычисляются значения двух близко лежащих точек $x_1 = x_{\text{ср}} - \varepsilon$ и $x_2 = x_{\text{ср}} + \varepsilon$, где ε наперед заданное малое значение аргумента функции (рис. 4.3). Далее сравниваются значения функции в точках x_1 и x_2 .

Исходя из условия существования только одного минимума функции на данном интервале поиска, можно определить, какую часть интервала поиска можно исключить на следующей итерации: при выполнении условия $f(x_1) < f(x_2)$ (см. рис. 4.3, а) – левую, при $f(x_1) > f(x_2)$ (см. рис. 4.3, б) – правую. Соответствующим образом

переносятся границы интервала $a \rightarrow x_{cp}$ или $b \rightarrow x_{cp}$ для продолжения поиска на следующей итерации. Поиск завершается, когда интервал поиска сокращается до наперед заданной малой величины δ . Возможна ситуация, когда минимум функции окажется лежащим в интервале $[x_1, x_2]$. В этом случае весь алгоритм поиска сведется к стягиванию интервала поиска к средней точке x_{cp} . Количество обращений к вычислению целевой функции определяется как $2N$, где N – число шагов поиска.

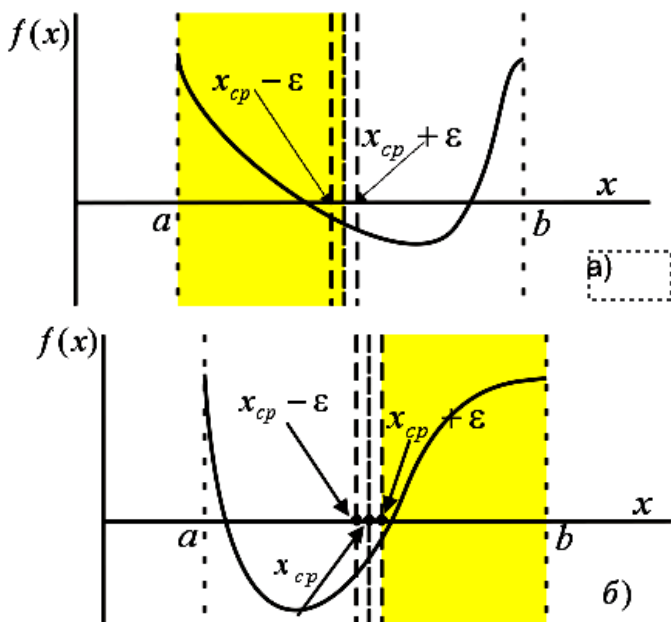


Рис. 4.3. Схема сокращения интервала поиска в зависимости от расположения минимума функции

Алгоритм поиска. Пусть функция $f(x)$ задана на отрезке $[a, b]$ и имеет только один минимум. Параметры ϵ, δ – малые значения.

Шаг 1. Определить середину интервала $x_{cp} = (a + b)/2$, выделить две точки на оси аргументов $x_1 = x_{cp} - \epsilon$ и $x_2 = x_{cp} + \epsilon$.

Шаг 2. Вычислить значения функций $f(x_1)$ и $f(x_2)$.

Шаг 3. Сравнить значения функции $f(x_1)$ и $f(x_2)$.

Если $f(x_1) < f(x_2)$, то присваиваем $b = x_{cp}$, иначе $a = x_{cp}$.

Шаг 4. Признак завершения счета. Если $abs(b - a) < \delta$, то перейти к шагу 5, иначе перейти к шагу 1.

Шаг 5. Вычислить $x_{\min} = (a + b)/2$ и $f(x_{\min})$.

Деление отрезка пополам. Метод, в котором на каждой итерации интервал поиска минимума функции сокращается вдвое. Он основан на выборе трех пробных точек, расположенных равномерно на интервале поиска минимума целевой функции. На текущей итерации определяется $x_{\text{ср}}$ середина отрезка $[a, b]$ и значения двух симметрично расположенных точек $x_1 = x_{\text{ср}} - abs(b - a)/4$ и $x_2 = x_{\text{ср}} + abs(b - a)/4$. Далее сравниваются значения функции в точках $x_{\text{ср}}$, x_1 и x_2 . Можно определить, какую часть интервала поиска можно исключить на следующей итерации.

На рис. 4.4 приведены возможные положения расположения минимума целевой функции, которые могут возникнуть в процессе сокращения интервала поиска. Заштрихованные области исключаются из дальнейшего рассмотрения. При $f(x_1) > f(x_{\text{ср}})$ – правую половину (см. рис. 4.4, а), при $f(x_1) < f(x_{\text{ср}})$ – левую половину (см. рис. 4.4, б) и при $f(x_2) > f(x_{\text{ср}})$ – левую и правую четверти (см. рис. 4.4, в). Соответствующим образом переносятся границы интервала поиска $a \rightarrow x_{\text{ср}}$ (см. рис. 4.4, а) или $b \rightarrow x_{\text{ср}}$ (см. рис. 4.4, б) или $a \rightarrow x_1$ и $b \rightarrow x_2$ (см. рис. 4.4, в) и формируется очередной интервал $[a, b]$. Поиск завершается, когда интервал поиска $[a, b]$ сокращается до заданной малой величины δ .

На первом шаге поиска производится три обращения к вычислению целевой функции в точках $x_{\text{ср}}$, x_1 и x_2 на последующих шагах поиска – два обращения (в точках x_1 и x_2). Общее количество обращений к вычислению целевой функции определяется как $(2N + 1)$, где N – число шагов поиска.

Алгоритм поиска. Пусть функция $f(x)$ задана на отрезке $[a, b]$ и имеет только один минимум, δ – малое значения.

Шаг 1. Определить середину интервала $x_{\text{ср}} = (a + b)/2$. Вычислить в этой точке значение функции $fx_{\text{ср}} = f(x_{\text{ср}})$.

Шаг 2. Определить точки $x_1 = x_{\text{ср}} - abs(b - a)/4$ и $x_2 = x_{\text{ср}} + abs(b - a)/4$.

Шаг 3. Вычислить значения функции $fx_1 = f(x_1)$, $fx_2 = f(x_2)$.

Шаг 4. Сравнить значения функции fx_1 и $fx_{\text{ср}}$.

Если $fx_1 < fx_{\text{ср}}$, то присвоить $b = x_{\text{ср}}$, $x_{\text{ср}} = x_1$, $fx_{\text{ср}} = fx_1$, вычислить точки $x_1 = x_{\text{ср}} - abs(b - a)/4$ и $x_2 = x_{\text{ср}} + abs(a)/4$, перейти к шагу 6. Иначе перейти к шагу 5.

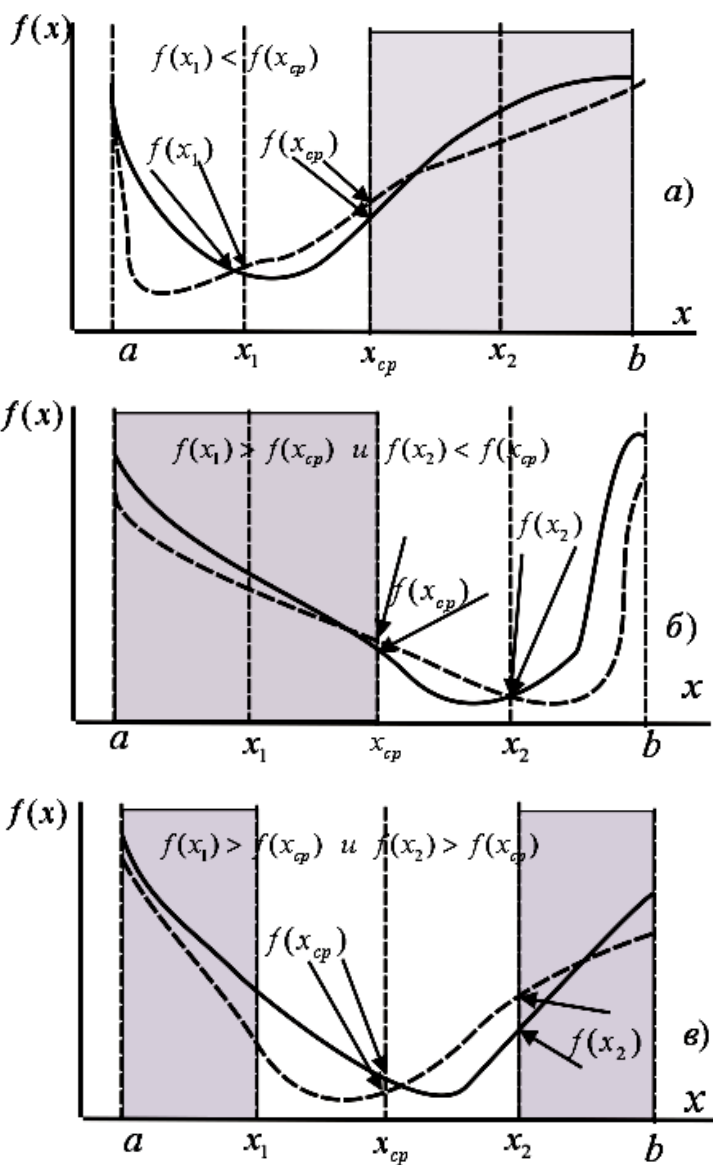


Рис. 4.4. Возможные положения минимумов целевой функции

Шаг 5. Сравнить значения функции fx_2 и fx_{cp} .

Если $fx_2 < fx_{cp}$, то присвоить $a = x_{cp}$, $x_{cp} = x_2$, $fx_{cp} = fx_2$, вычислить точки $x_1 = x_{cp} - abs(b - a)/4$ и $x_2 = x_{cp} + abs(b - a)/4$, перейти к шагу 6. Иначе присвоить $a = x_1$, $b = x_2$, вычислить точки $x_1 = x_{cp} - abs(b - a)/4$ и $x_2 = x_{cp} + abs(b - a)/4$.

Шаг 6. Завершения счета. Если $abs(b - a) < \delta$, то перейти к шагу 7, иначе перейти к шагу 3.

Шаг 7. Вычислить $x_{min} = (a + b)/2$ и $f(x_{min})$.

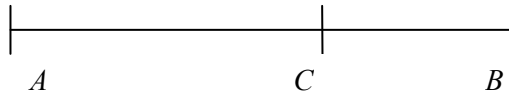
Золотое сечение. Из рассмотрения ранее приведенных методов можно сделать выводы, позволяющие повысить эффективность поиска.

1. Если количество пробных точек равно двум, то их следует размещать на одинаковых расстояниях от середины интервала.

2. В соответствии с общей минимаксной стратегией пробные точки должны размещаться по симметричной схеме таким образом, чтобы отношение длины исключаемого подынтервала к величине интервала поиска оставалось постоянным.

3. На каждой итерации поиска должна вычисляться только одна пробная точка и значение целевой функции в этой точке.

Геометрическая интерпретация принципа метода золотого сечения заключается в следующем. Дана прямая ACB :



Отношение отрезков прямой определяется соотношением $AC/AB = CB/AC$, т.е. отношение большего отрезка к целому равно отношению меньшего отрезка к большему отрезку. Если принять целый отрезок $AB = 1$, то $AC = (-1 + \sqrt{5})/2$. Приблизленно $AC \approx 0,62$, а $CB \approx 0,38$.

Процесс поиска минимума целевой функции носит итерационный характер, и положение пробных точек относительно текущего размера интервала поиска на каждой итерации переопределяется. Рассмотрим исходное симметричное расположение двух пробных точек на единичном интервале (рис. 4.5). Пробные точки отстоят от краев интервала на расстоянии λ (под λ понимается не абсолютная, а относительная величина для текущего интервала поиска). При

таким симметричным расположении точек длина оставшегося подынтервала (после исключения ненужной для дальнейшего анализа части) всегда равна λ независимо от того, какое значение функции в пробных точках окажется меньшим. Предположим, что исключили правый подынтервал. Оставшийся подынтервал длины λ содержит одну пробную точку, расположенную на расстоянии $(1 - \lambda)$ от его левой граничной точки. Чтобы сохранить симметрию в стратегии поиска, расстояние $(1 - \lambda)$ предшествующего шага поиска должно составлять λ -ю часть длины нового интервала поиска (которая на текущем шаге уже составляет величину, равную λ от предшествующего интервала). При таком выборе следующая пробная точка расположится на расстоянии равном λ -й части длины интервала от правой граничной точки (см. рис. 4.5). Этот алгоритм реализуется при $\lambda = (-1 + \sqrt{5})/2$. Поиск, при котором интервал поиска делится пробными точками в указанном отношении, называется *поиском с помощью золотого сечения* (см. рис. 4.5). На первой итерации целевая функция вычисляется в двух пробных точках, на последующих итерациях производится однократное обращение к вычислению целевой функции.

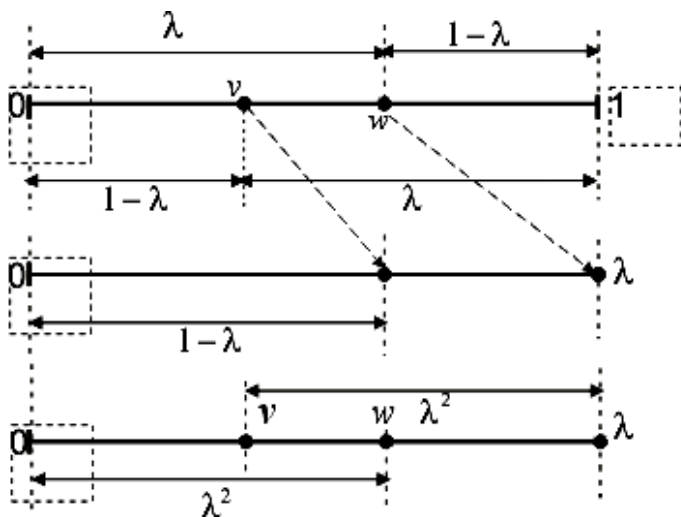


Рис. 4.5. Поиск с помощью метода золотого сечения.
Симметрия золотого сечения интервала

Количество обращений к вычислению целевой функции определяется как $(N + 1)$, где N – число шагов поиска.

Пусть функция $f(x)$ задана на отрезке $[a, b]$ и имеет только один минимум, параметр δ – малое значение.

Шаг 1. Задать коэффициенты $\lambda_2 = 0,62$, $\lambda_1 = 0,38$, интервал $[a, b]$ и длину отрезка $l = abs(b - a)$.

Шаг 2. Выделить две точки $v = a + l\lambda_1$ и $w = a + l\lambda_2$.

Шаг 3. Если $abs(b - a) < \delta$, то перейти к шагу 7, иначе перейти к шагу 4.

Шаг 4. Вычислить в этих точках значения функции $f_v = f(v)$ и $f_w = f(w)$.

Шаг 5. Сравнить значения функции f_w и f_v .

Если $f_w < f_v$, то присвоить $a = v$, $v = w$, $f_v = f_w$ и вычислить $w = a + abs(b - a)\lambda_2, f(w)$. Перейти к шагу 3, иначе перейти к шагу 6.

Шаг 6. Присвоить $b = w$, $w = v$, $f_w = f_v$ и вычислить $v = a + abs(b - a)\lambda_1, f(v)$. Перейти к шагу 3.

Шаг 7. Вычислить $x_{\min} = (a + b)/2$ и $f(x_{\min})$.

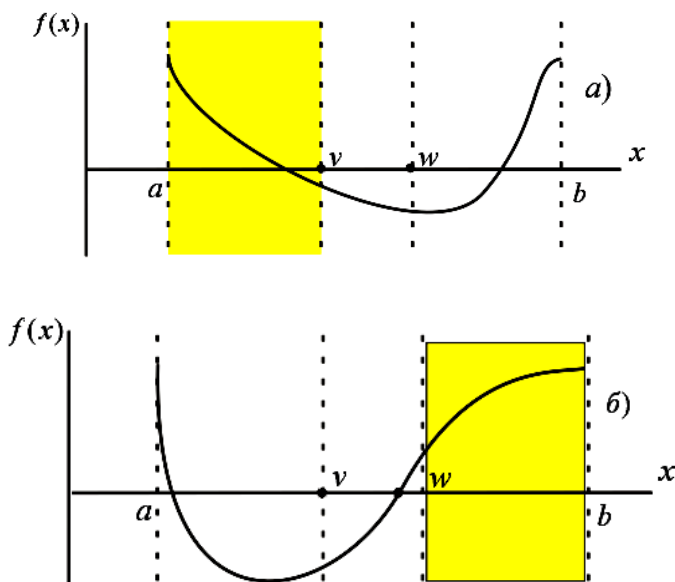


Рис. 4.6. Возможные положения минимума целевой функции

Пусть длина исходного интервала поиска L_1 , а конечного интервала – L_n . Эффективность метода можно оценить по отношению L_n/L_1 . В методах дихотомии и деления отрезка пополам длина конечного интервала составляет $L_1(0,5)^{N/2}$, а в методе золотого сечения – $L_1(0,618)^{N-1}$. При заданной точности вычислений ε количество обращений к вычислению целевой функции в двух первых методах составляет $N = 2 \ln(\varepsilon) / \ln(0,5)$, а для метода золотого сечения $N = 1 + \ln(\varepsilon) / \ln(0,618)$.

Метод Фибоначчи. В методе золотого сечения значение параметра λ является фиксированным. Метод Фибоначчи в принципе ничем не отличается от метода золотого сечения, за исключением того, что параметр λ является вычисляемым и определяется, исходя из наперед заданного допустимого количества обращений к вычислению целевой функции m . Следовательно, если количество вычислений целевой функции m задано априори, то для этого варианта поиска параметр λ сокращения интервала поиска определяется однозначно. Пусть $\lambda = \lambda_m$, тогда алгоритм последовательного сокращения интервала поиска $[a, b]$ будет выглядеть следующим образом

$$\begin{array}{lll}
 & \Delta_0 & \Delta_0 = b_0 - a_0 \\
 k = 1 & \Delta_1 = \Delta_0 \cdot \lambda_m & \Delta_1 = b_1 - a_1 \\
 k = 2 & \Delta_2 = \Delta_0 - \Delta_1 = \Delta_0 \cdot (1 - \lambda_m) & \Delta_2 = b_2 - a_2 \\
 k = 3 & \Delta_3 = \Delta_1 - \Delta_2 = \Delta_0 \cdot (-1 + 2 \cdot \lambda_m) & \Delta_3 = b_3 - a_3 \\
 k = 4 & \Delta_4 = \Delta_2 - \Delta_3 = \Delta_0 \cdot (2 - 3 \cdot \lambda_m) & \Delta_4 = b_4 - a_4 \\
 k = 5 & \Delta_5 = \Delta_3 - \Delta_4 = \Delta_0 \cdot (-3 + 5 \cdot \lambda_m) & \Delta_5 = b_5 - a_5 \\
 k = 6 & \Delta_6 = \Delta_4 - \Delta_5 = \Delta_0 \cdot (5 - 8 \cdot \lambda_m) & \Delta_6 = b_6 - a_6 \\
 k = i & &
 \end{array} \quad (4.12)$$

$$\Delta_i = \Delta_{i-2} - \Delta_{i-1} = \Delta_0 \cdot (-1)^i (F_{i-1} - F_i \lambda_m) \quad \Delta_i = b_i - a_i \quad (4.13)$$

где F_i – числа Фибоначчи.

Принцип построения последовательности чисел Фибоначчи (табл. 4.1)

$$F_{i+2} = F_{i+1} + F_i. \quad (4.14)$$

Таблица 4.1

F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	...	F_i	...
1	1	2	3	5	8	13	21	34	55	...		...

Так же как и в методе золотого сечения, пробные точки на k -й итерации будем обозначать v_k и w_k .

В зависимости от анализа значений целевой функции в пробных точках v и w границы интервала поиска формируются как

$$a_k = a_{k-1}, \quad w_k = v_{k-1}, \quad b_k = w_{k-1},$$

или находится новое значение v_k и $f(v_k)$, т.е.

$$b_k = b_{k-1}, \quad v_k = w_{k-1}, \quad a_k = v_{k-1}.$$

Вычисляется новое значение w_k и $f(w_k)$.

При предельном числе обращений к целевой функции, равном m , последний интервал поиска можно выразить как

$$\Delta_m = (-1)^m \Delta_0 (F_{m-1} - \lambda_m F_m). \quad (4.15)$$

Если положить $\lambda_m = F_{m-1} / F_m$, интервал поиска $\Delta_m = 0$.

Числа Фибоначчи F_k можно вычислить для любого значения индекса k , используя формулу Бинэ:

$$F_k = \frac{1}{\sqrt{5}} \left[\left(\frac{1+\sqrt{5}}{2} \right)^k - \left(\frac{1-\sqrt{5}}{2} \right)^k \right], \quad k=1,2,\dots \quad (4.16)$$

Пусть

$$\lambda_{k+1} = F_k / F_{k+1}, \quad k=1, 2, \dots \quad (4.17)$$

Определяем параметр

$$\lambda_{k+1} = \frac{\sqrt{5}-1}{2} \frac{1-(-1)^k \left(\frac{3-\sqrt{5}}{2} \right)^k}{1-(-1)^{k+1} \left(\frac{3-\sqrt{5}}{2} \right)^{k+1}}, \quad (4.18)$$

используя формулу Бинэ, в зависимости от фиксированного значения индекса k .

При $k \rightarrow \infty$ $\lim_{k \rightarrow \infty} \lambda_k = \frac{\sqrt{5}-1}{2}$. Следовательно, параметр сокращения интервала поиска в методе Фибоначчи (табл. 4.2) в пределе

совпадает с параметром сокращения интервала поиска в методе золотого сечения. Предпоследний интервал поиска $\Delta_{m-1} = b_{m-1} - a_{m-1}$ определяет точность вычисления. Последний шаг выполняется с использованием алгоритма дихотомии.

В табл. 4.3 приведено относительное уменьшение интервала.

Таблица 4.2

k	1	2	3	4	5	6
λ_k	1	0,5	0,666667	0,6	0,625	0,615385

Окончание табл. 4.2

k	7	8	9	10	11	12
λ_k	0,61905	0,61765	0,61818	0,61798	0,61806	0,61803

Таблица 4.3

Число вычислений функции	Деление отрезка пополам	Метод дихотомии	Метод золотого сечения	Метод Фибоначчи
1	1,0	1,0	1,0	1,0
2	—	0,5	0,618	0,5
3	0,5	—	0,382	0,333
4	—	0,25	0,236	0,200
5	0,25	—	0,146	0,125
6	—	0,125	0,090	0,077
7	0,125	—	0,056	0,048
8	—	0,0625	0,0345	0,0294
9	0,0625	—	0,0213	0,0182
10	—	0,0312	0,0132	0,0112
11	0,0312	—	0,00813	0,00694
12	—	0,0156	0,00502	0,00429
13	0,0156	—	0,00311	0,00265
14	—	0,00781	0,00192	0,00164
15	0,00781	—	0,00119	0,00101
16	—	0,00391	0,000733	0,000626
17	0,00391	—	0,000453	0,000387
18	—	0,00195	0,000280	0,000239
19	0,00195	—	0,000173	0,000148
20	—	0,000976	0,000107	0,0000913

Исследование методов сокращения интервала поиска минимума целевой функции показало, что наибольшая эффективность достигается при симметричном расположении пробных точек относи-

тельно краев интервала. При большом числе обращений к вычислению целевой функции методы золотого сечения и Фибоначчи практически совпадают по эффективности. Различие между ними заключается в том, что метод Фибоначчи применим для случая, когда количество обращений к вычислению целевой функции заранее фиксировано. В методе золотого сечения этого ограничения не существует. Следует отметить, что приближенные вычисления коэффициента λ приводят к накоплению ошибки, что, в свою очередь, может привести к нарушению принципа вложенности отрезков $[b_k, a_k]$ и расходимости процесса поиска.

Рассмотренные алгоритмы являются базовыми и допускают различные модификации.

4.2.2. Полиномиальная аппроксимация (метод с использованием квадратичной аппроксимации, методы с использованием производных, метод поиска с использованием кубической аппроксимации)

Рассматриваемые методы [12], в отличие от методов исключения интервалов, применимы к гладким функциям. Идея методов состоит в аппроксимации гладкой функции полиномом, для которого можно определить точку оптимума. Необходимыми условиями являются унимодальность и непрерывность исследуемой функции. Эти методы логично использовать для поиска минимума одномерной функции в тех случаях, когда направления поиска связаны с вычислением производных целевой функции. Эта ситуация имеет место и при решении задач многомерной оптимизации, когда определяется минимум целевой функции по выбранному направлению спуска. Качество оценки координаты точки оптимума, полученной с помощью аппроксимирующего полинома, можно повысить двумя способами – использовать полиномы более высокого порядка или уменьшить интервал аппроксимации. Второй способ предпочтительней, так как построение полинома высокого порядка может оказаться сложной процедурой. На практике ограничиваются полиномами до третьего порядка.

Метод с использованием квадратичной аппроксимации. Простейшим вариантом полиномиальной аппроксимации является интерполяция квадратичным полиномом (линейная интерполяция

дает минимальные значения на краях интервала). В качестве интерполирующей функции можно использовать квадратичный полином $p(x)$, построенный по первой интерполяционной формуле Ньютона:

$$p(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2), \quad (4.19)$$

где x_i, y_i – узлы и значения табличной функции в узлах; $h = (x_{i+1} - x_i)$ – шаг таблицы.

Коэффициенты полинома определяются как

$$a_0 = y_1, \quad a_i = \Delta^i y_1 / (i! \cdot h^i), \quad (4.20)$$

$\Delta^i y_1$ – конечная разность.

Таким образом, по трем заданным точкам (значениям функции в узлах) можно построить квадратичный полином и по первой производной определить минимум интерполяционной функции.

$$\bar{x} = (x_2 + x_1) / 2 - (a_1 / 2a_2). \quad (4.21)$$

Поскольку рассматриваемая функция и интерполирующий полином являются унимодальными функциями, то можно ожидать, что $\bar{x}$ окажется приемлемой координатой для истинного минимума x^* .

На этом принципе построен алгоритм Пауэлла [15].

Начальная точка x_1 , h – шаг по оси x .

Шаг 1. Вычислить $x_2 = x_1 + h$.

Шаг 2. Вычислить $f(x_1)$ и $f(x_2)$.

Шаг 3. Если $f(x_1) > f(x_2)$, положить $x_3 = x_1 + 2h$. Если $f(x_1) \leq f(x_2)$, положить $x_3 = x_1 - h$.

Шаг 4. Вычислить $f(x_3)$ и найти

$$f_{\min} = \min \{f_1, f_2, f_3\},$$

$x_{\min}$ – точка x_i , которая соответствует $f_{\min}$.

Шаг 5. По точкам x_1, x_2, x_3 вычислить $\bar{x}$, используя процедуру интерполяции.

Шаг 6. Проверка на окончание поиска. Если $f_{\min} - f(\bar{x})$ и $x_{\min} - \bar{x}$ оказываются достаточно малыми, то поиск закончен. Иначе перейти к шагу 7.

Шаг 7. Выбрать «наилучшую» точку $x_{\min}$ или $\bar{x}$ и две точки по обе стороны от нее, пронумеровав их в естественном порядке, и перейти к шагу 4.

Примечание. При первой реализации шага 5 границы интервала, содержащего точку минимума, необязательно оказываются установленными. Полученная точка $\bar{x}$ может находиться за точкой x_3 . Для того, чтобы исключить возможность слишком большого экстраполяционного перемещения, следует провести после шага 5 дополнительную проверку и в случае, когда точка $\bar{x}$ находится слишком далеко от x_3 , заменить $\bar{x}$ точкой, координата которой вычисляется заранее установленной длиной шага.

Методы с использованием производных. Если рассматриваемые целевые функции являются не только непрерывными, но и дифференцируемыми, то эффективность процедур поиска можно повысить.

Метод Ньютона – Рафсона. Пусть функция $f(x)$ дважды дифференцируема, тогда для функции $f'(x) = \phi(x)$ условием нахождения минимума (экстремума) является нахождение такого x^* , для которого $\phi'(x^*) = 0$. Решение этого уравнения производится методом касательных. Процесс носит итерационный характер. Выберем в качестве стартовой точки значение (первое приближение значения корня уравнения $\phi'(x) = 0$) x_k . Из точки A (рис. 4.7) проведем касательную до пересечения с осью абсцисс в точке x_{k+1} . Уравнение касательной имеет вид

$$\widehat{\phi}(x) - \phi(x_k) = \phi'(x_k)(x - x_k), \quad (4.22)$$

где x – текущее значение координаты, а $\widehat{\phi}(x)$ – соответствующее значение линейной функции. Полагая $\widehat{\phi}(x) = 0$, получаем

$$x_{k+1} = x_k - \frac{\phi(x_k)}{\phi'(x_k)}. \quad (4.23)$$

Вычислим значение функции $\phi(x_{k+1})$ – точка B (второе приближенное значение корня). Из точки B проведем касательную до пересечения с осью абсцисс в точке x_{k+2} . Процесс продолжается до тех пор, пока не реализуется условие $|\phi(x_{k+i})| < \varepsilon$ (ε – малое число).

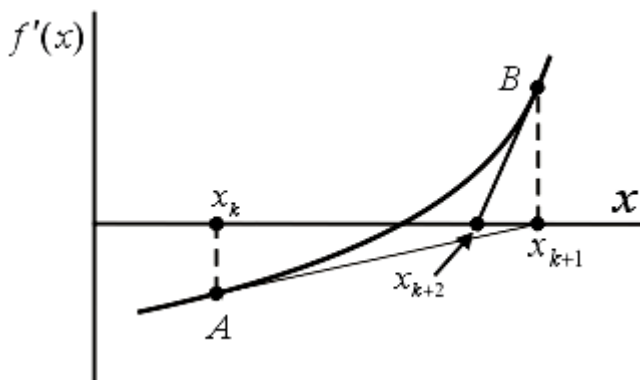


Рис. 4.7. Процесс поиска минимума $f'(x) = \varphi(x)$

В этом алгоритме большое значение имеет выбор начальной точки. При неудачном выборе процесс поиска корня может расходиться. Рассмотренный алгоритм используется в разделе численных методов для решения нелинейных и трансцендентных уравнений. В поиске минимума функции $\varphi(x)$ можно использовать и метод секущих (хорд).

Метод поиска с использованием кубической аппроксимации. Метод по своей сути схож с методом, использующим квадратичную аппроксимацию, но для построения аппроксимирующего полинома третьего порядка требуется меньшее число точек, так как в каждой точке вычисляется не только значение целевой функции, но и ее производная. Начальная точка x_1 выбирается произвольно, точка x_2 выбирается таким образом, чтобы производные $f'(x_1)$ и $f'(x_2)$ имели разные знаки. В этом случае в интервале $[x_1, x_2]$ будет находиться точка x^* , содержащая минимум функции ($f'(x^*) = 0$). Аппроксимирующий полином третьего порядка запишется в виде

$$p(x) = a_0 + a_1(x - x_1) + a_2(x - x_1)(x - x_2) + a_3(x - x_1)^2(x - x_2). \quad (4.24)$$

Параметры этого уравнения подбираются так, чтобы значения $p(x)$ и $p'(x)$ в точках x_1 и x_2 совпадали со значениями $f(x)$ и $f'(x)$ в этих же точках. Выполнение этих условий приводит к нахождению коэффициентов a_0, a_1, a_2, a_3 из решения системы уравнений

$$\begin{aligned}
f(x_1) &= a_0, \\
f(x_2) &= a_0 + a_1(x_2 - x_1), \\
f'(x_1) &= a_1 + a_2(x_1 - x_2), \\
f'(x_2) &= a_1 + a_2(x_1 - x_2) + a_3(x_2 - x_1)^2.
\end{aligned} \tag{4.25}$$

После нахождения коэффициентов, приравнявая первую производную полинома $p(x)$ нулю, можно получить выражение для определения точки минимума.

$$x^* = \begin{cases} x_2, & \text{если } \mu < 0; \\ x_2 - \mu(x_2 - x_1), & \text{если } 0 \leq \mu \leq 0; \\ x_1, & \text{если } \mu > 1, \end{cases} \tag{4.26}$$

где

$$\begin{aligned}
\mu &= \frac{f'(x_2) + w - z}{f'(x_2) - f'(x_1) + 2w}, \\
z &= \frac{3(f(x_1) - f(x_2))}{x_2 - x_1} + f'(x_2) + f'(x_1), \\
w &= \begin{cases} (z^2 - f'(x_2) \cdot f'(x_1))^{1/2}, & \text{если } x_1 < x_2, \\ -(z^2 - f'(x_2) \cdot f'(x_1))^{1/2}, & \text{если } x_1 > x_2. \end{cases}
\end{aligned} \tag{4.27}$$

Получаемая точка x^* расположена между точками x_1 и x_2 . Затем снова выбираются две точки, одна из которых – x^* , а другая – либо x_1 , либо x_2 (значения производных в паре выбранных точек должны быть противоположны по знаку). Далее процесс кубической аппроксимации продолжается до достижения необходимой точности определения минимума.

Алгоритм метода кубической аппроксимации может быть следующим.

Выбрать начальную точку x_0 , h – шаг по переменной, ε_1 и ε_2 – параметры сходимости.

Шаг 1. Вычислить $f'(x_0)$.

Если $f'(x_0) < 0$, то вычислить $x_{K+1} = x_K + 2^K h$, $K = 0, 1, 2, \dots$

Если $f'(x_0) > 0$, то вычислить $x_{K+1} = x_K - 2^K h$, $K = 0, 1, 2, \dots$

Шаг 2. Вычислить $f'(x)$ в точках x_{K+1} , $K = 0, 1, 2, \dots$, вплоть до точки x_m , в которой $f'(x_{m-1}) \cdot f'(x_m) < 0$.

Затем положить $x_1 = x_{m-1}$, $x_2 = x_m$, вычислить $f(x_1)$, $f'(x_1)$, $f(x_2)$ и $f'(x_2)$.

Шаг 3. Найти точку x^* аппроксимирующего полинома (по формуле (4.26)).

Шаг 4. Если $f(x) < f(x_1)$, перейти к шагу 5. Иначе вычислять x^* по формуле $x^* = x^* \cdot \frac{1}{2}(x^* - x_1)$ до тех пор, пока не будет выполнено условие $f(x^*) \leq f(x_1)$.

Шаг 5. Условие окончания поиска.

Если $|f(x^*)| \leq \varepsilon_1$ и $|(x^* - x_1) / x^*| \leq \varepsilon_2$, то поиск завершен.

Иначе положить, либо

$$x_2 = x_1, \quad x_1 = x^*, \quad \text{если} \quad f'(x^*) \cdot f'(x_1) < 0,$$

либо

$$x_1 = x^*, \quad \text{если} \quad f'(x^*) \cdot f'(x_2) < 0.$$

Затем перейти к шагу 3.

В случае, когда значения производной вычисляются непосредственно, рекомендуется использовать метод кубической аппроксимации. Если производная вычисляется методом численного дифференцирования, то эффективней использовать метод Пауэлла. Все вышеприведенные методы поиска минимума целевой функции допускают различного рода модификации, позволяющие улучшить его быстродействие или повысить точность вычислений. Выбор нужного алгоритма является своего рода искусством и зависит от характерных особенностей решаемой задачи.

4.3. Поиск минимума функций многих переменных

Разделение оптимизационных задач на классы удобно проводить по свойствам целевой функции и функциям ограничения. Целевая функция может быть квадратичной, полиномом, сепарабельной, однородной, составной, дробной и т.д. Функции ограничения могут быть нелинейными, квадратичными, полиномами, линейными и т.п.

Ограничения могут быть односторонними, двусторонними или вообще отсутствовать. Используя различные сочетания видов целевой функции и ограничений, можно выделить различного типа задачи безусловной оптимизации и различные типы задач с ограничениями (условная оптимизация). Методы поиска минимума целевой функции различаются способами построения базовых и пробных точек в пространстве поиска, определением направления и способом движения к минимуму целевой функции.

В настоящее время существует большое число базовых методов оптимизации функций многих переменных и их модификаций. Рассмотрим наиболее известные и часто употребляемые методы, связанные с безусловной оптимизацией (т.е. без ограничений) целевых функций многих переменных, с самых общих положений.

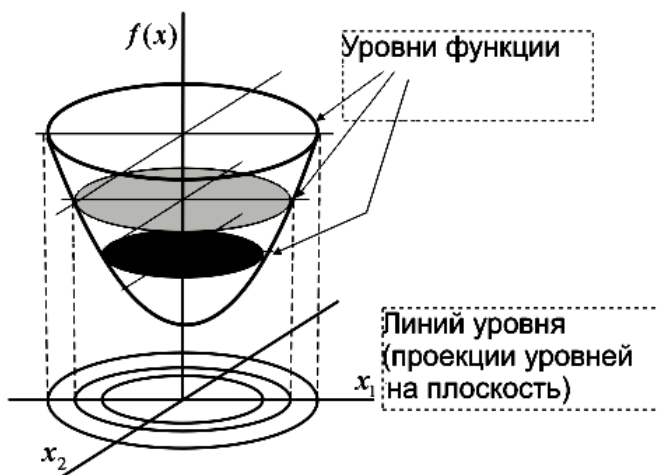


Рис. 4.8. Линии уровня целевой функции

В дальнейшем рассмотрении используется понятие «линии уровня целевой функции». На рис. 4.8 показано несколько уровней простой целевой функции (параболоид вращения). Каждый уровень характеризуется одним значением целевой функции. Верхний обод параболоида, а также его сечения (затененные эллипсы) проецируются на плоскость $\{x_1, x_2\}$. Эти проекции являются линиями уровня (линиями, каждая точка которых имеет одно и то же чис-

ленное значение соответствующего уровня целевой функции). Подобным способом в картографии указываются глубины морей и высота местности. При рассмотрении задач многомерной оптимизации могут быть полезны некоторые геометрические представления. Целевая функция представляется в виде так называемой поверхности отклика. В одномерном случае (целевая функция – функция одной переменной) она вырождается в кривую на плоскости. В двумерном случае целевую функцию можно представить в трехмерном пространстве (как, например, на рис. 4.8). При большей размерности пространства поиска минимума целевой функции поверхности отклика являются гиперповерхностями, которые невозможно отобразить обычными средствами.

Запишем вектор x , определенный в евклидовом пространстве R^n через его компоненты $x = (x^1, x^2, \dots, x^n)$. Функцию многих переменных можно разложить в ряд Тейлора

$$f(x) = f(x^*) + \nabla f(x^*)^T \Delta x + 1/2 \Delta x^T \nabla^2 f(x^*)^T \Delta x + O_3(x), \quad (4.28)$$

где x^* – точка разложения функции из пространства R^n ; $\Delta x = x - x^*$ – величина изменения x ; $\nabla f(x)$ – n -мерный вектор-столбец первых производных функции $f(x)$, вычисленных в точке x^* ; $\nabla^2 f(x^*) = H_j(x^*)$ – симметрическая матрица размерности $n \times n$ вторых частных производных функции $f(x)$, вычисленных в точке x^* (матрица Гессе).

Пренебрегая в (4.28) членами высших порядков, представим целевую функцию как квадратичную функцию, которая занимает значительное место в дальнейшем рассмотрении материала. Проведем анализ возможного поведения целевой функции в окрестности ее экстремума.

Для произвольного значения x изменение целевой функции можно записать как

$$\Delta f(x) = f(x) - f(x^*) = \nabla f(x^*)^T \Delta x + 1/2 \Delta x^T \nabla^2 f(x^*)^T \Delta x. \quad (4.29)$$

В окрестности точки минимума выполняется условие

$$\Delta f(x) = f(x) - f(x^*) \geq 0. \quad (4.30)$$

Если условие (4.30) выполняется для всех $x \in R^n$, то обозначим $x^* \rightarrow x^{**}$ как точку *глобального* минимума функции многих переменных. Если это условие справедливо только в окрестности x^* , эта точка соответствует *локальному* минимуму.

Если $\Delta f \leq 0$, то имеет место локальный максимум, если Δf принимает положительные, отрицательные значения или равна нулю в δ -окрестности, то x^* – *седловая* точка.

В предположении, что $f(x)$, $\nabla f(x)$ и $\nabla^2 f(x)$ существуют и непрерывны для всех $x \in R^n$, знак Δf не меняется при произвольном варьировании Δx только в случае $\Delta f(x^*) = 0$.

Точка x^* является *стационарной*. Следовательно, условие стационарности

$$\Delta f(x^*) = 0 \quad (4.31)$$

и уравнение (4.29) принимает вид

$$\Delta f(x) = 1/2 \Delta x^T \nabla^2 f(x^*)^T \Delta x. \quad (4.32)$$

Знак $\Delta f(x)$ определяется квадратичной формой

$$Q(x) = 1/2 \Delta x^T \nabla^2 f(x^*)^T \Delta x \quad (4.33)$$

или $Q(z) = z^T A z$.

Стационарная точка x^* – точка минимума, если матрица A является положительно полуопределенной матрицей, т.е. $Q(z) \geq 0$ для любых z .

Существование минимума функции нескольких переменных в точке x^* определяется условиями необходимости и достаточности. Чтобы точка x^* являлась точкой минимума функции нескольких переменных *необходимо*, чтобы

$$\Delta f(x^*) = 0 \quad (4.34)$$

и матрица $\nabla^2 f(x^*)$ была положительно полуопределенная, и *достаточно*, чтобы

$$\Delta f(x^*) = 0 \quad (4.35)$$

и матрица $\nabla^2 f(x^*)$ была положительно определенной (т.е. $Q(z) > 0$), рис. 4.9.

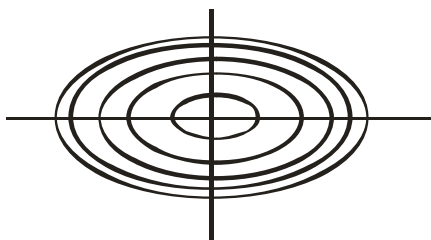


Рис. 4.9. Линии уровня «чаши», матрица $Q(z)$ положительно определена

При выполнении условий необходимости и достаточности точка x^* является точкой изолированного (строгого) локального минимума $f(x)$. Поскольку априори, в общем случае, неизвестно поведение $f(x)$ во всей области поиска минимума, приходится ограничиваться нахождением локального минимума (рис. 4.10). Но если можно показать, что $x^T \nabla^2 f(x^*)^T x \geq 0$ для всех x , то $f(x)$ является выпуклой функцией и найденный минимум является глобальным.

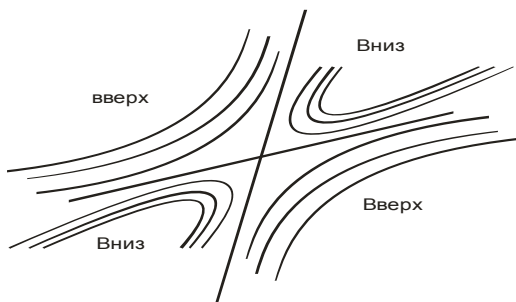


Рис. 4.10. Линии уровня «седла», матрица $Q(z)$ неопределена положительно

На рис. 4.10 показан случай, когда матрица $Q(z)$ неопределена положительно, а на рис. 4.11 – когда матрица вырождена.

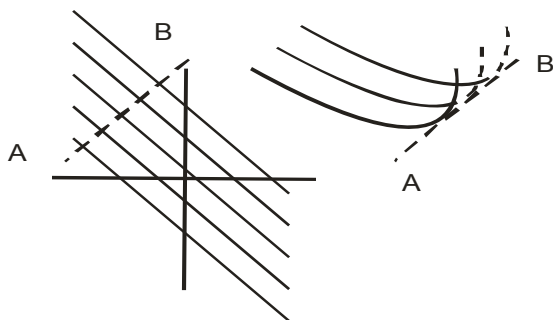


Рис. 4.11. Линии уровня «оврага», матрица $Q(z)$ вырождена.
Целевая функция квадратична относительно линии $A - B$

Задачи, решаемые с использованием методов оптимизации функции многих переменных, весьма разнообразны по своей сути, и поэтому каждому решению должен соответствовать адекватный математический метод. Методы решения задач безусловной оптимизации можно разделить на несколько широких классов:

- 1) методы прямого поиска, построенные только на вычислении значений целевой функции;
- 2) градиентные методы, в которых используются точные значения первых производных целевой функции;
- 3) методы второго порядка, в которых используются первые и вторые производные целевой функции;
- 4) стохастические методы.

Перед написанием программы, реализующей любой алгоритм поиска, полезно рассмотреть возможности преобразования задачи с целью улучшения поведения целевой функции в области минимума, сокращения времени вычисления функции и ее градиента, уменьшения влияния погрешностей вычисления на процесс оптимизации. Каких-либо универсальных рекомендаций не существует, и каждый случай требует индивидуального подхода.

Преобразование переменных. Бывает полезным произвести замену переменных с целью изменить характер линий уровня (поверхностей уровня в многомерном случае), придавая им менее изогнутый характер и вытягивая вдоль координаты. В зависимости от конкретного случая замены могут носить как линейный, так и нелинейный характер.

Масштабирование целевой функции. Решение не изменится, если целевую функцию умножить на некоторую положительную константу или добавить константу, чтобы значения целевой функции находились в области, близкой к единице. Эта операция называется *масштабированием целевой функции*, которая непосредственно связана с преобразованием переменных. Размерности компонент вектора аргумента целевой функции x могут оказать существенное влияние на анализ условий окончания поиска $|x^{(i+1)} - x^{(i)}| < \delta$ – сравнение двух соседних итераций поиска – i -й и $(i + 1)$ -й. Если размерности координат различаются на порядки, то определяющим фактором будет координата с наибольшими значениями.

Масштабирование ограничений. Изменение масштаба ограничений должно согласовываться с принципом преобразования переменных и масштабированием целевой функции.

Независимо от выбранного метода, необходимо решить задачи:

- 1) выбор базовой (стартовой) точки алгоритма;
- 2) выбор направления поиска минимума целевой функции;
- 3) выбор способа движения по направлению к минимуму.

4.3.1. Методы прямого поиска (симплекс-метод, методы спуска, метод покоординатного спуска, метод Хука-Дживса)

Симплекс-метод. Метод, применяемый для решения задач нелинейного программирования, никак не связан с симплексным методом линейного программирования, кроме созвучности названий. Он относится к методам, основанным на вычислениях только значений целевой функции. Прототипом этого метода являлся метод эволюционной оптимизации, стратегия поиска в котором заключается в следующем – задается произвольная базовая точка в многомерной области поиска и исследуются определенным образом выделенные близлежащие к ней другие точки. Из этой группы точек выбирается точка с наилучшим (наименьшим) значением целевой функции и процесс повторяется. Подобная стратегия проста, но в случае большого числа переменных (большой размерности пространства поиска) требует больших вычислительных ресурсов и далеко не эффективна. Развитие подобной, но более эффективной стратегии, реализовано в методе поиска по симплексу. Основу это-

го метода составляет регулярный симплекс, который представляет собой многогранник, ребра которого – прямые линии, пересекающиеся в $(n + 1)$ -й равноотстоящих друг от друга вершинах, определенный в n -мерном пространстве. Например, в случае двумерного пространства симплексом является равносторонний треугольник (рис. 4.12), а трехмерном – тетраэдр.

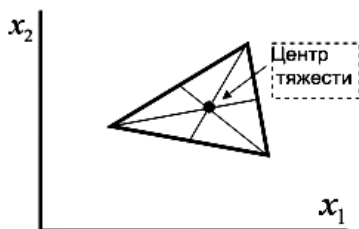


Рис. 4.12. Пример симплекса в двумерном пространстве

Процедура поиска начинается с построения начального симплекса в произвольной базовой точке пространства. Следует отметить, выбор месторасположения остальных вершин многогранника определяет размер симплекса. В выбранных вершинах симплекса вычисляются значения целевой функции.

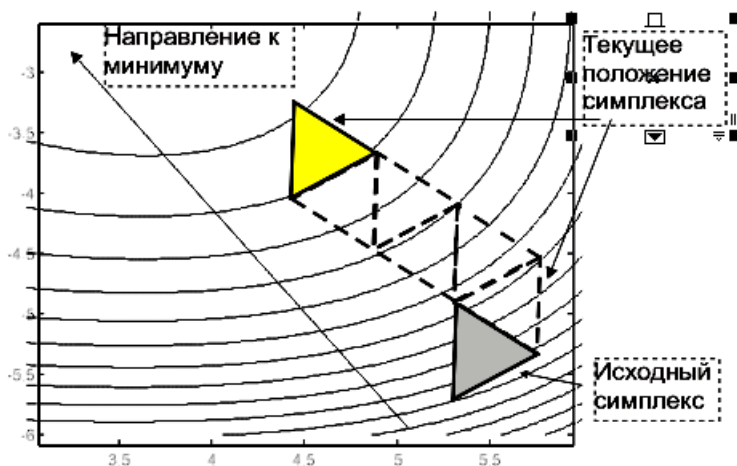


Рис. 4.13. Процесс трансформации симплекса

Выбирается вершина, в которой значение целевой функции наибольшее (наихудшая вершина). Определяется центр тяжести симплекса и прямая, соединяющая наихудшую вершину и центр тяжести симплекса. По направлению, определяемому этой прямой, строится вершина нового симплекса (рис. 4.13). Итерационный процесс продолжается до тех пор, пока очередной симплекс не захватит точку минимума целевой функции или не произойдет «зацикливание» алгоритма – циклическое движение по двум соседним положениям симплексов. Разрешение этой проблемы заключается в следующих действиях.

1. Если в новой полученной вершине значение целевой функции окажется худшим, то осуществляется возврат к вершине прежнего симплекса. Вместо нее берется другая вершина, которой соответствует следующее по величине значение целевой функции и отыскивается точка, которая является ее дополнением (на соответствующей ей прямой).

2. Если какая-либо вершина симплекса не исключается на протяжении более чем m итераций, то необходимо уменьшить размеры симплекса с помощью коэффициента редукции и построить новый симплекс, выбрав опорной точкой вершину симплекса, имеющую наименьшее значение целевой функции (рис. 4.14).



Рис. 4.14. Редукция симплекса

Значение m выбирается из соотношения

$$m = 1,65n + 0,05n^2, \quad (4.36)$$

где n – размерность пространства поиска; m округляется до ближайшего целого числа.

3. Поиск завершается, когда размеры симплекса или разности между значениями целевой функции в вершинах симплекса станут достаточно малыми. Степень малости задается начальными условиями, определяющими требуемую точность решения задачи.

Построение симплекса начинается с задания координат базовой точки $x^{(0)}$ и масштабного множителя α . Верхний индекс $(\cdot)$ означает номер базовой точки или вершины симплекса. Координаты остальных вершин симплекса определяются как

$$x^{(i)} = \begin{cases} x_j^{(0)} + \delta_1 & \text{для } j \neq i; \\ x_j^{(0)} + \delta_2 & \text{для } j = i, \end{cases} \quad (4.37)$$

$$i, j = 1, 2, 3, \dots, n.$$

Величины приращений координат δ_1 и δ_2 , зависящие от n и α , определяются по формулам

$$\delta_1 = \left[\frac{(n+1)^{1/2} + n - 1}{n\sqrt{2}} \right] \alpha; \quad \delta_2 = \left[\frac{(n+1)^{1/2} - 1}{n\sqrt{2}} \right] \alpha. \quad (4.38)$$

Величина α выбирается исходя из характеристик решаемой задачи. При $\alpha = 1$ ребра регулярного симплекса имеют единичную длину.

Центр тяжести симплекса

$$x_c = \frac{1}{n} \sum_{\substack{i=0 \\ i \neq j}}^n x^{(i)}. \quad (4.39)$$

Все точки прямой, проходящие через центр тяжести симплекса и вершину $x^{(j)}$ симплекса, лежат на прямой

$$x = x^{(j)} + \lambda(x_c - x^{(j)}). \quad (4.40)$$

Исходной точке $x^{(j)}$ соответствует значение $\lambda = 0$, центру тяжести x_c соответствует $\lambda = 1$. Новая вершина регулярного симплекса получается при $\lambda = 2$ и определяется соотношением

$$x_{\text{нов}}^{(j)} = 2x_c - x_{\text{предыдущ}}^{(j)}. \quad (4.41)$$

Рассмотренный алгоритм поиска минимума целевой функции симплекс-методом является базовым алгоритмом подобного типа, который обладает своими достоинствами и недостатками.

К достоинствам можно отнести следующие свойства.

1. Алгоритм достаточно прост с точки зрения написания программы расчета.

2. Алгоритм не требует большого объема памяти – хранится информация о координатах $(n + 1)$ -й вершине симплекса.

3. Алгоритм малочувствителен к ошибкам вычисления значений целевой функции, так как при его реализации оперируют с наибольшими значениями целевой функции в вершинах симплекса, а не с наименьшими.

К недостаткам следует отнести следующее.

1. Алгоритм достаточно медленный, так как информация, полученная на предшествующих итерациях, никак не используется на текущей итерации.

2. В алгоритме не предусмотрен простой способ расширения симплекса, не требующего пересчета значений целевой функции во всех вершинах образца. Уменьшение размеров симплекса может происходить при движении в узком «овраге» или по «хребту» поверхности, отражающей целевую функцию. При выходе из «оврага» (или «хребта») дальнейший поиск ведется в условиях уменьшенного шага.

3. Возможны трудности с масштабированием переменных. Желательно все переменные отмасштабировать таким образом, чтобы они имели сопоставимые по величине значения.

Модификацией симплекс-метода является алгоритм Нелдера-Мида, который позволяет при отражении симплекса осуществлять операции его сжатия и растяжения (т.е. отказ от сохранения свойств регулярности симплекса в процессе поиска).

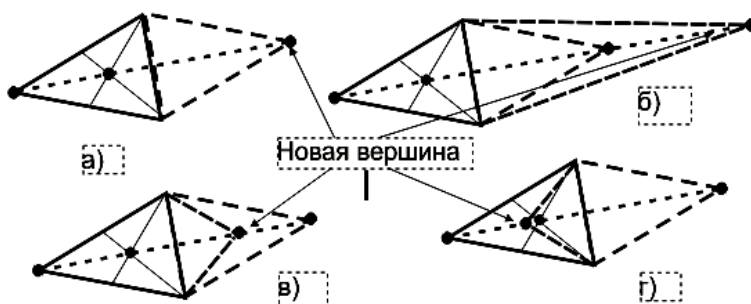


Рис. 4.15. Преобразование симплекса в методе Нелдера-Мида

В методе Нелдера-Мида преобразование симплекса – растяжение, сжатие и редукция – производятся на основании анализа трех вершин симплекса (двумерная задача). Первая вершина $x^{(\max)}$, в которой значение целевой функции $f(x^{(\max)})$ наибольшее (наихудшая вершина). Вторая вершина $x^{(2)}$, в которой целевая функция $f(x^{(2)})$ имеет второе по величине значение (после $f(x^{(\max)})$). Третья вершина $x^{(\min)}$, в которой значение целевой функции $f(x^{(\min)})$ наименьшее (наилучшая вершина). Отражение вершины симплекса производится вдоль прямой

$$x = x^{(\max)} + \lambda(x_c - x^{(\max)}),$$

или

$$x = x^{(\max)} + (1 + \theta)(x_c - x^{(\max)}). \quad (4.42)$$

Эти уравнения прямой эквивалентны уравнению (4.40). Параметр θ определяет характер преобразования симплекса.

При $f(x^{(\min)}) < f(x^{(\text{нов})}) < f(x^{(2)})$ и $\theta = 1$ происходит нормальное отражение вершины симплекса (см. рис. 4.15, а).

При $f(x^{(\text{нов})}) < f(x^{(\min)})$ и $\theta > 1$ – растяжение симплекса (см. рис. 4.15, б).

При $f(x^{(\text{нов})}) > f(x^{(2)})$, $f(x^{(\text{нов})}) \geq f(x^{(\max)})$ и $\theta < 0$ – сжатие симплекса (см. рис. 4.15, в).

При $f(x^{(2)}) < f(x^{(\text{нов})}) < f(x^{(\max)})$ и $\theta > 0$ – сжатие симплекса (см. рис. 4.15, г).

Рекомендуется в качестве удовлетворительных значений параметра θ принять следующие величины: $\theta = 1$ – нормальное отражение; $\theta = 0,5$ – сжатие; $\theta = 2$ – растяжение.

Последовательность действий следующая.

Шаг 1. Задать значения параметра преобразования симплекса θ и δ – точность вычислений.

Шаг 2. Построить исходный симплекс.

Шаг 3. Вычислить x_c , $x^{(\max)}$, $x^{(\min)}$, $x^{(2)}$.

Шаг 4. Произвести анализ целевой функции в точках $x^{(\max)}$, $x^{(\min)}$, $x^{(2)}$ и вычислить положение новой вершины, используя при-

веденные выше условия преобразования симплекса и уравнение прямой (4.42).

Шаг 5. Сравнить значения целевой функции в вершинах симплекса на двух последовательных итерациях – если значения различаются на величину меньшую или равную δ , то закончить поиск, иначе перейти к шагу 3.

Паркинсон и Хатчинсон [12] исследовали влияние значений коэффициента преобразования симплекса и способа построения исходного симплекса на эффективность поиска. Они установили, что ориентация исходного симплекса влияет на эффективность поиска значительно сильнее, чем его размеры, и предложили выбрать значения коэффициента преобразования симплекса: $\theta = 2$ – нормальное отражение; $\theta = 0,25$ – сжатие; $\theta = 2,5$ – растяжение. Ориентация симплекса в пространстве влияет на выбор направления поиска (4.42) и, следовательно, быстроту достижения минимума.

В литературе имеются рекомендации по увеличению числа вершин симплекса $(n + k)$, $k = 1, 2, \dots$, так как при сложной конфигурации линей уровня возможно вырождение симплекса. Например, в двумерном случае симплекс может вырождаться в прямую. Наличие дополнительных вершин позволяет избежать этой опасности, но приводит к увеличению времени расчета.

Методы спуска. Различные методы спуска, применяемые для решения задач оптимизации, различаются либо выбором направления спуска, либо способом движения вдоль направления спуска или тем и др.

В области поиска задается начальная точка

$$x^{(0)} = (x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, \dots, x_n^{(0)})$$

и выбирается направление спуска, на котором определяется точка $x^{(1)}$, выбирается новое направление и определяется точка $x^{(2)}$ и т.д. (n – размерность пространства). В результате строится последовательность точек $x^{(i)}$, сходящая к точке минимума целевой функции.

Выбор направления. Во всех методах, независимо от способа определения направления спуска, должно соблюдаться условие

$f(x^{(i)}) > f(x^{(i+1)})$, т.е. значение целевой функции на последующей итерации меньше, чем на предыдущей итерации.

Метод покоординатного спуска. Метод покоординатного спуска относится к методам нулевого порядка, в которых процесс поиска базируется только на вычислении значений целевой функции в выбранных соответствующим образом пробных точках. Существует много вариантов метода покоординатного спуска. Простейшим алгоритмом, реализующим этот метод, является метод циклического изменения переменных, при котором каждый раз изменяется только одна переменная. В качестве направлений поиска выбираются координатные направления. Вдоль каждого из координатных направлений последовательно производится поиск минимума целевой функции с использованием описанных ранее методов поиска минимума одномерной функции.

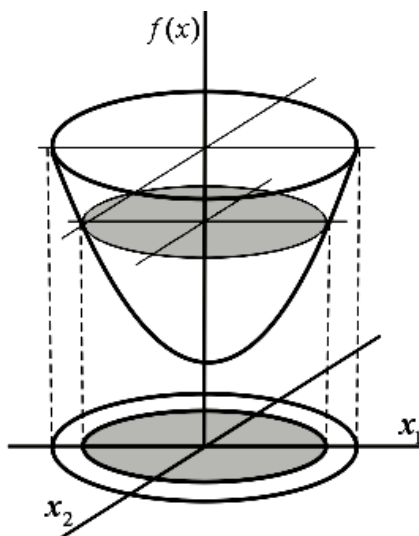


Рис. 4.16. Двумерная целевая функция, проекция линий уровня на плоскость $\{x_1, x_2\}$

На рис. 4.16 двумерная функция – параболоид, построенный в координатах $\{x_1, x_2, f(x)\}$. При более сложных целевых функциях

линии уровня могут иметь весьма сложную конфигурацию. На рис. 4.17 представлено сечение двумерной функции. Параболоид расsection вертикальной плоскостью $A-A'-B'-B$, параллельной оси x_2 (значение x_1 фиксировано). На поверхности параболоида сечение представлено параболой $A-C-B$, проекцией которой на плоскость $\{x_1, x_2\}$ является прямая $A'-C'-B'$. Минимум параболы находится в точке C . На прямой $A'-C'-B'$ точке минимума соответствует точка C' .

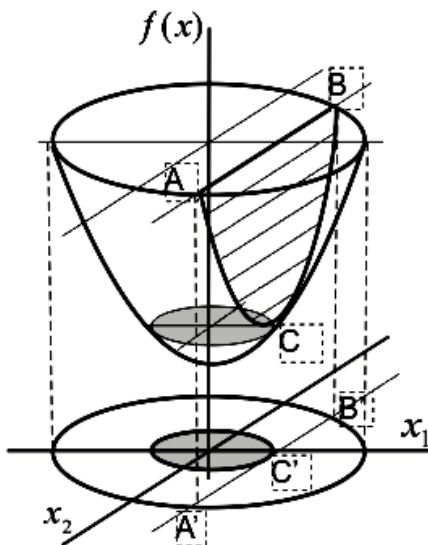


Рис. 4.17. Сечение целевой функции

На рис. 4.18 представлен алгоритм поиска минимума двумерной функции. В заданной области задается стартовая точка A и по линии 1-1 (параллельной координатной оси x_1) при фиксированном значении x_2 определяется точка B , в которой значение целевой функции по этому направлению минимально. Затем из точки B по линии 2-2 (параллельной координатной оси x_2) при фиксированном значении x_1 , найденном на предшествующем поиске, определяется точка C , в которой значение целевой функции по этому на-

правлению минимально. Поскольку в данном примере размерность пространства равна двум, данная итерация завершается. Из точки C производится выполнение нового итерационного циклического перебора координатного поиска и так до тех пор, пока значения целевой функции, полученные на двух соседних итерациях, не будут отличаться друг от друга на заданную ранее малую величину.



Рис. 4.18. Поиск минимума двумерной функции методом покоординатного спуска

Подобный алгоритм хорошо работает, если целевая функция обладает свойством сферической симметрии (или достаточно близка к ней). При сложной конфигурации линий уровня конечный результат поиска может оказаться зависимым от порядка перебора координатных направлений. Однако если линии уровня сильно искривлены и имеют сложную конфигурацию, то алгоритм может превратиться в бесконечную последовательность итераций («зациклится»).

Рассмотрим по шагам работу алгоритма на примере двумерной функции (см. рис. 4.18).

Шаг 1. Выбрать начальную точку A с координатами $(x_1^{(0)}, x_2^{(0)})$.

Запомнить $x_{\text{предш}} = x^{(0)}$.

Шаг 2. Организовать циклический просмотр переменных. Зафиксировать значение $x_2^f = x_2^{(0)}$, тогда целевая функция $f(x_1, x_2)$

будет функцией одной переменной $f(x_1)$ с параметром x_2^f . Найти минимум $f(x_1)$ по направлению 1-1 любым методом одномерной оптимизации (точка B) с координатами $(x_1^{(1)} = x_1^f, x_2^f)$.

Зафиксировать значение x_1^f , тогда целевая функция $f(x_1, x_2)$ будет функцией одной переменной $f(x_2)$ с параметром x_1^f . Найти минимум $f(x_2)$ по направлению 2-2 любым методом одномерной оптимизации (точка C) с координатами $(x_1^f, x_2^2 = x_2^f)$. (В случае многомерной функции эти процедуры повторяются для каждого координатного направления.) Итерация заканчивается определением промежуточного минимума $(x_1^{(i)} = x_1^f, x_2^{(i)} = x_2^f)$.

Шаг 3. Проверить условия достижения минимума – значения целевой функции $f(x_1, x_2)$, т.е. $\|x^f - x^{\text{предш}}\| < \delta$.

Если «Да», то закончить поиск, иначе положить $x_1^{(0)} = x_1^f, x_2^{(0)} = x_2^f, x^{\text{предш}} = x^f$ и перейти к шагу 2.

Метод Хука-Дживса. Этот метод является модификацией метода покоординатного спуска, компенсирующей некоторые его недостатки. Суть изменений заключается в том, что помимо покоординатного спуска периодически проводится спуск по направлению $d = x^{(i+1)} - x^{(i)}$, которое определяется точками промежуточного определения минимума целевой функции на двух последовательных итерациях поиска. Следовательно, в отличие от метода циклического перебора координатных направлений, здесь для определения направления спуска используется информация, полученная на предшествующих итерациях. Метод Хука-Дживса является комбинацией исследующего поиска, циклического изменения переменных и поиска по образцу (рис. 4.19).

Исследующий поиск заключается в последовательности следующих действий:

- задаются величины шагов по всем координатным направлениям (они могут быть различными);
- задается исходная точка;
- по одной из координат делается пробный шаг (если значение целевой функции в пробной точке не превышает ее значения в ис-

ходной точке, то шаг удачный; в противном случае из исходной точки делается шаг в обратном направлении);

- перебор осуществляется по всем координатам (полученная точка называется *базовой*).

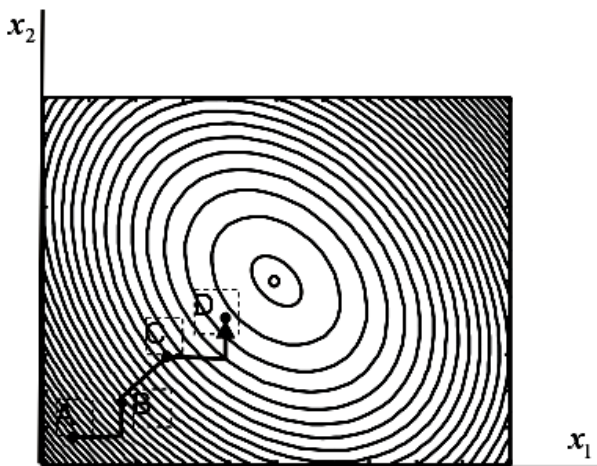


Рис. 4.19. Поиск минимума целевой функции методом Хука-Дживса

Поиск по образцу. В данном алгоритме

$$x_p^{(i+1)} = x^{(i)} + (x^{(i)} - x^{(i-1)}) \quad (4.43)$$

приняты следующие обозначения: $x^{(i)}$ – текущая базовая точка; $x^{(i-1)}$ – предыдущая базовая точка; $x_p^{(i+1)}$ – точка, построенная при движении по образцу; $x^{(i+1)}$ – следующая (новая) базовая точка.

В алгоритме производятся следующие действия.

Производится единственный шаг вдоль прямой соединяющей две базовые точки – полученную и предыдущую. Новая базовая точка на этом направлении определяется по формуле (4.43).

Как только движение по образцу не приводит к уменьшению значения целевой функции, точка $x_p^{(i+1)}$ фиксируется в качестве временной базовой точки и относительно нее проводится исследующий поиск. Если в результате поиска определяется точка, в ко-

торой значение целевой функции меньше чем в точке $x^{(i)}$, то она рассматривается как новая базовая точка $x^{(i+1)}$.

Если исследующий поиск неудачен, то нужно вернуться в точку $x^{(i)}$ и провести исследующий поиск с целью определения нового направления минимизации.

Если в конечном итоге (т.е. после просмотра всех возможных направлений) поиск не приводит к успеху, то нужно уменьшить величину шага и повторить исследующий поиск.

Поиск завершен, когда величина шага становится достаточно малой.

Рассмотрим алгоритм Хука-Дживса по шагам.

Шаг 1. Задать следующие параметры:

начальную точку $x^{(0)}$;

приращения по координатам Δ_k , $k = 1, 2, 3, \dots, n$;

коэффициент уменьшения шага $\alpha > 1$;

параметр окончания поиска (малое значение) $\varepsilon > 0$.

Шаг 2. Провести исследующий поиск в текущей базовой точке, начиная с $x^{(0)}$.

Шаг 3. Проверить результаты исследующего поиска на успешный результат:

успешный – перейти к шагу 5;

неуспешный – перейти к шагу 4.

Шаг 4. Проверить условие окончания поиска – если величина шага $\|\Delta x\| < \varepsilon$, то прекратить поиск, последняя базовая точка является точкой минимума.

Иначе уменьшить приращение по координатам

$$\Delta_k = \Delta_k / \alpha, \quad k = 1, 2, 3, \dots, n,$$

и перейти к шагу 2.

Шаг 5. Провести поиск по образцу $x_p^{(i+1)} = x^{(i)} + (x^{(i)} - x^{(i-1)})$.

Шаг 6. Провести исследующий поиск, используя $x_p^{(i+1)}$ в качестве базовой точки и считать, что найденная $x^{(i+1)}$ – следующая (новая) базовая точка.

Шаг 7. Проверить условие $f(x^{(i+1)}) < f(x^{(i)})$. Если условие – «Истина», то положить $(x^{(i-1)} = x^{(i)})$, $(x^{(i)} = x^{(i+1)})$ и перейти к шагу 5. Иначе перейти к шагу 4.

Алгоритм имеет недостатки:

- при сложной конфигурации линий уровня на этапе циклического движения по координатам возможно преждевременное прекращение поиска;
- при значительных нелинейных эффектах возможно вырождение – последовательность исследующих поисков не переходит в движение по образцу.

Метод имеет различные модификации, связанные исследующим поиском, выбором направления и длины шага.

4.3.2. Метод сопряженных направлений. Метод сопряженных направлений Пауэла

Наиболее эффективным из методов прямого поиска является метод сопряженных направлений, который, так же как и методы других видов поиска, имеет различные реализуемые алгоритмы. Основным достоинством этого метода является то, что при работе алгоритма для построения направлений поиска используется информация, полученная на предшествующей итерации. Метод ориентирован на решение задач с квадратичными целевыми функциями.

В общем виде квадратичную функцию запишем как

$$q(x) = a + b^T x + \frac{1}{2} x^T Q x. \quad (4.44)$$

Рассмотрим понятие сопряженных направлений. Пусть Q – симметрическая матрица размерности $n \times n$.

Направления $d^{(1)}, d^{(2)}, d^{(3)}, \dots, d^{(r)}, r \leq n$, называются *сопряженными* (Q -сопряженными), если эти направления линейно независимы и

$$(d^{(i)})^T Q d^{(j)} = 0 \quad (4.45)$$

для всех $i \neq j$.

Если задана произвольная точка $x^{(1)}$, положение точек, находящихся на направлении d задается формулой

$$x = x^{(1)} + \lambda d. \quad (4.46)$$

Минимум функции $q(x)$ (4.44) вдоль направления d определяется нахождением такого значения λ^* , при котором $\partial q / \partial \lambda = 0$. Следовательно, полагая, что минимум функции находится в точке $y^{(1)}$, можно записать

$$\left[\left(y^{(1)} \right)^T Q + b^T \right] d = 0. \quad (4.47)$$

Для другой произвольной точки $x^{(2)}$ минимум функции находится в точке $y^{(2)}$, можно записать

$$\left[\left(y^{(2)} \right)^T Q + b^T \right] d = 0. \quad (4.48)$$

Вычитая (4.47) из (4.48), получим

$$\left(y^{(2)} - y^{(1)} \right)^T Q d = 0. \quad (4.49)$$

Это означает, что направления d и $\left(y^{(2)} - y^{(1)} \right)$ являются Q -сопряженными. Данное построение отражает *свойство параллельного подпространства* для квадратичных функций. На рис. 4.20 показаны сопряженные направления для двумерного случая. Видно, что для квадратичной функции можно построить систему сопряженных направлений и найти точку минимума на основании трех одномерных поисков.

В рассмотренном алгоритме для построения системы сопряженных направлений использовались начальные точки и некоторое направление. Это может быть не всегда удобно, поэтому лучше строить сопряженные направления, исходя из одной начальной точки.

Процедура построения заключается в следующем (например, для двумерного случая).

Пусть алгоритм начинается из точки $x^{(0)}$ (рис. 4.21) и в дальнейшем построении используются единичные координатные векторы $e^{(1)}, e^{(2)}$ (в многомерном случае $e^{(1)}, e^{(2)}, e^{(3)}, \dots, e^{(n)}$).

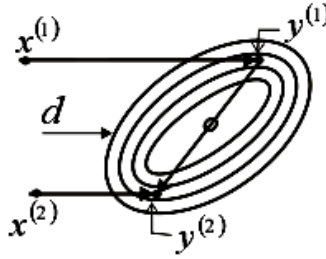


Рис. 4.20. Сопряженные направления на плоскости

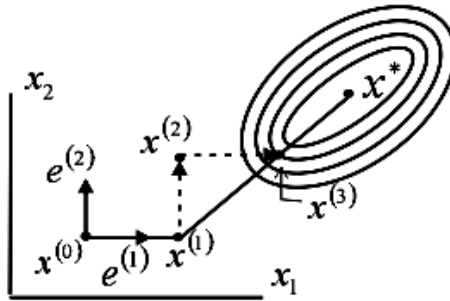


Рис. 4.21. Поиск минимума методом сопряженных направлений

Для заданной начальной точки $x^{(0)}$ вычисляется значение $\lambda^{(0)}$, которому соответствует минимальное значение функции $f(x^{(0)} + \lambda^{(0)} e^{(1)})$. Определим

$$x^{(1)} = x^{(0)} + \lambda^{(0)} e^{(1)} \quad (4.50)$$

и вычислим $\lambda^{(1)}$, которому соответствует минимальное значение функции $f(x^{(1)} + \lambda^{(1)} e^{(2)})$.

Определим

$$x^{(2)} = x^{(1)} + \lambda^{(1)} e^{(2)}. \quad (4.51)$$

Далее вычислим $\lambda^{(2)}$, минимизирующее $f(x^{(2)} + \lambda^{(2)} e^{(1)})$, и положим

$$x^{(3)} = x^{(2)} + \lambda^{(2)} e^{(1)}. \quad (4.52)$$

Направления $(x^{(3)} - x^{(1)})$ и $e^{(1)}$ оказываются сопряженными. Этот принцип можно применить и к n -мерному пространству. В этом случае для нахождения минимума квадратичной функции требуется проведение n^2 одномерных поисков.

Метод сопряженных направлений Пауэла.

Шаг 1. Задать начальную точку $x^{(0)}$ и систему n линейно независимых направлений $d^{(i)}$; возможен вариант, когда эти направления совпадают с единичными координатными векторами $d^{(i)} = e^{(i)}, i = 1, 2, \dots, n$.

Шаг 2. Найти минимум функции $f(x)$ при последовательном движении по $(n+1)$ направлению. При первом поиске – начальная, а при последующих проходах полученная ранее точка минимума берется в качестве исходной точки. Направление $d^{(n)}$ используется как на первом, так и на последнем поиске.

Шаг 3. Определить новое сопряженное направление с помощью обобщенного свойства параллельного подпространства (см. ранее).

Шаг 4. Заменить направление $d^{(1)}$ на $d^{(2)}$ и т.д. Заменить $d^{(n)}$ полученным сопряженным направлением. Перейти к шагу 2.

Для квадратичной функции алгоритм приводит к нахождению точки минимума за n циклов. Если целевая функция – не квадратичная, то для нахождения минимума требуется более чем n циклов (после прохождения n циклов требуется обновление метода). Алгоритм чувствителен к ошибкам счета, так как они могут привести к нарушению свойств ортогональности.

4.3.3. Градиентные методы (метод Коши, метод Ньютона)

Рассмотренные прямые методы поиска для получения решения требуют очень большое количество вычислений целевой функции. Кроме того, в них никак не используются свойства стационарных точек (т.е. точек, удовлетворяющих необходимому условию первого порядка – формула (4.31)). Эти два обстоятельства привели к необходимости рассмотрения методов поиска, связанных с градиентом целевой функции. Предполагается, что компоненты градиен-

та могут быть вычислены либо аналитически, либо при помощи численных методов с высокой степенью точности. Поскольку антиградиент функции указывает направление, по которому функция убывает наиболее быстро, то, естественно, спуск нужно проводить в этом направлении. Все градиентные методы носят итеративный характер, и связь между итерациями определяется соотношением

$$x^{(i+1)} = x^{(i)} + \alpha^{(i)} d(x^{(i)}), \quad (4.53)$$

где $x^{(i)}$ – текущее приближение (итерация) к решению x^* ; $\alpha^{(i)}$ – параметр, определяющий длину шага в направлении поиска $d^{(i)} = d(x^{(i)})$ в n -мерном пространстве переменных $x_j, j = 1, 2, \dots, n$.

Необходимо решить вопрос о выборе $\alpha^{(i)}$ и направлении $d(x^{(i)})$.

Выбор $\alpha^{(i)}$ определяется методами одномерной оптимизации при минимизации функции $f(x)$ в выбранном направлении. Направление $d(x^{(i)})$ можно определить различными методами. Оно непосредственно связано с градиентом целевой функции. И одной из разновидностей метода выбора направления $d(x^{(i)})$ являются методы наискорейшего спуска.

Метод Коши. Пусть в некоторой произвольной точке $\bar{x}$ многомерного пространства нужно определить направление локального наискорейшего спуска, т.е. направление, по которому в окрестности точки $\bar{x}$ происходит наискорейшее уменьшение целевой функции. Процедура Коши основана на линейной аппроксимации. Разлагая функцию в ряд Тейлора и ограничиваясь первыми двумя членами ряда, можно записать

$$f(x) = f(\bar{x}) + \nabla f(\bar{x})^T \Delta x. \quad (4.54)$$

Локальное уменьшение функции определяется вторым членом, так как значение первого члена фиксировано. Следовательно, наибольшее уменьшение целевой функции происходит в таком направлении (4.53), которому соответствует наибольшая отрицательная величина второго члена уравнения (4.54). Из свойств скалярного

го умножения следует, что выбор нужного направления произойдет при условии

$$d(\bar{x}) = -\nabla f(\bar{x}), \quad (4.55)$$

и второй член примет вид

$$-\alpha \nabla f(\bar{x})^T \nabla f(\bar{x}). \quad (4.56)$$

Поэтому в основе простейшего градиентного метода лежит формула

$$x^{(i+1)} = x^{(i)} - \alpha^{(i)} \nabla f(x^{(i)}), \quad (4.57)$$

где $\alpha^{(i)}$ – заданный положительный параметр, вычисленный вдоль направления $\nabla f(x^{(i)})$ методом одномерного поиска.

Метод Коши требует вычисления целевой функции и ее первых производных на каждой итерации. Следует отметить, что он хорошо работает, когда линии уровня являются окружностями или близки к ним и направление антиградиента, вообще говоря, не может служить глобальным направлением поиска минимума. На рис. 4.22 представлен алгоритм Коши для двумерного случая. В точке $x^{(0)}$ вычисляется градиент целевой функции, и по направлению антиградиента методами одномерной оптимизации определяется минимум – точка $x^{(1)}$. Далее процедура повторяется и определяется глобальный минимум в точке $x^{(2)}$. Если бы линии уровня были окружностями, то уже на первой итерации можно было бы определить положение глобального минимума двумерной функции, так как линия антиградиента проходила бы через точку $x^{(2)}$.

Поскольку определение минимума по направлению производится численными методами с определенной погрешностью, то для достижения глобального минимума с нужной степенью точности возможно повторение итерационных циклов поиска.

Метод имеет ряд достоинств и недостатков. К недостаткам следует отнести необходимость определять $\alpha^{(i)}$ на каждой итерации, а также медленную сходимость к точке минимума из-за малых значений градиента в окрестности минимума. Достоинствами метода

являются его устойчивость и хорошее продвижение из точек расположенных далеко от минимума.

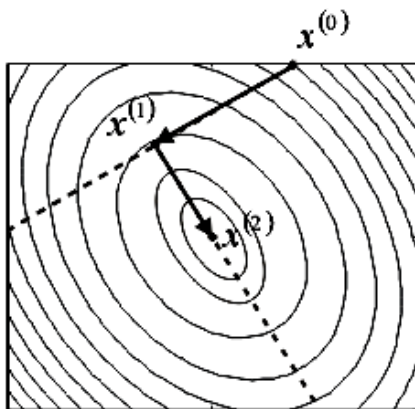


Рис. 4.22. Итерации поиска по методу Коши

Рассмотрим принципиальный алгоритм, который завершает работу при выполнении одного из условий:

- либо предельные значения модуля градиента меньше ε_1 ;
- либо предельные значения модуля вектора смещения Δx между соседними итерациями — ε_1 ;
- либо число итераций превышает предельное значение m .

Шаг 1. Задать начальную точку $x^{(0)}$, задать размерность пространства поиска n , задать предельные значения для модуля градиента и модуля вектора смещения Δx между соседними итерациями — ε_1 , задать точность определения минимума одномерного поиска — ε_2 ;

задать предельное число итераций m .

Положить $i = 0$.

Шаг 2. Вычислить градиент $\nabla f(x^{(i)})$.

Шаг 3. Проверить условие $\|\nabla f(x^{(i)})\| \leq \varepsilon_1$. Если «Да», то перейти к шагу 9, иначе перейти к шагу 4.

Шаг 4. Проверить условие $i \geq m$. Если «Да», то перейти к шагу 9, иначе перейти к шагу 5.

Шаг 5. Поиск минимума по направлению антиградиента одним из методов одномерной оптимизации (использовать значение ε_2).

Шаг 6. Положить $x^{(i+1)} = x^{(i)} - \alpha^{(i)} \nabla f(x^{(i)})$.

Шаг 7. Проверить условие $\frac{\|x^{(i+1)} - x^{(i)}\|}{\|x^{(i)}\|} \leq \varepsilon_1$. Если «Да», то пе-

рейти к шагу 9, иначе перейти к шагу 8.

Шаг 8. Положить $i = i + 1$ и перейти к шагу 2.

Шаг 9. Останов.

Метод Ньютона. В отличие от метода Коши метод Ньютона основан на квадратичной аппроксимации целевой функции (разложение в ряд Тейлора до членов второго порядка). В алгоритме используются вторые производные целевой функции:

$$\bar{f}(x; x^{(i)}) = f(x^{(i)}) + \nabla f(x^{(i)})^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(x^{(i)}) \Delta x, \quad (4.58)$$

где $\bar{f}(x; x^{(i)})$ – аппроксимирующая функция переменной x , построенной в точке $x^{(i)}$.

Процесс поиска минимума – итерационный процесс, на каждой итерации которого во вновь получаемой точке $x^{(i+1)}$ градиент аппроксимирующей функции $f(x)$ обращался в нуль.

Тогда

$$\nabla \bar{f}(x; x^{(i)}) = \nabla f(x^{(i)}) + \nabla^2 f(x^{(i)}) \Delta x = 0 \quad (4.59)$$

и, следовательно,

$$\Delta x = -\nabla^2 f(x^{(i)})^{-1} \nabla f(x^{(i)}). \quad (4.60)$$

Последовательность построения точек минимума на текущей итерации определяется соотношением

$$x^{(i+1)} = x^{(i)} - \nabla^2 f(x^{(i)})^{-1} \nabla f(x^{(i)}). \quad (4.61)$$

Последовательность шагов в алгоритме Ньютона такая же, как в алгоритме Коши, различается только процесс построения точек $x^{(i+1)}$ – сравним выражения (4.57) и (4.61).

4.3.4. Методы сопряженных градиентов (алгоритм метода сопряженных градиентов, алгоритм Флетчера-Ривса, алгоритм Дэвидона-Флетчера-Пауэла (с переменной метрикой))

Эти методы часто используются даже, когда целевая функция не является выпуклой. В этом случае определяется локальный минимум. Направление вектора градиента функции, определенного в точке x n -мерного пространства указывает направление возрастания этой функции. Следовательно, для достижения минимума целевой функции необходимо организовать движение в направлении, обратном градиенту функции.

Пусть функция $f(x)$ строго выпукла, дважды непрерывно дифференцируема и x^* – оптимальное решение. Тогда $\nabla f(x^*) = 0$. Обозначим вектор приращения аргумента как h . Уравнение (4.28) можно записать в виде

$$\begin{aligned} f(x) - f(x^*) &= \frac{1}{2} \langle x - x^*, H(x^*)(x - x^*) \rangle = \\ &= - \langle H(x^*)x^*, x \rangle + \frac{1}{2} \langle x, H(x^*)x \rangle + \frac{1}{2} \langle x^*, H(x^*)x^* \rangle, \end{aligned} \quad (4.62)$$

где $H(x^*)$ – симметрическая матрица размерности $n \times n$ вторых производных целевой функции $f(x)$, вычисленных в точке x^* .

Характерной чертой этих методов является положение, что минимум выпуклой функции может быть достигнут не более чем за n итераций. Это утверждение касается принципиального алгоритма, в реализуемом алгоритме для достижения минимума с заданной точностью возможно повторение циклов поиска. В конкретных разновидностях метода сопряженных градиентов это связано с погрешностями вычисления градиента, определения минимума по направлению и точности вычислений.

Алгоритм метода сопряженных градиентов.

Шаг 1. Выбрать точку $x_0 \in R^n$ и положить $i = 0$.

Шаг 2. Вычислить градиент $\nabla f(x_i)$.

Шаг 3. Если $\nabla f(x_i) = 0$, то поиск закончен. Иначе вычислить h_i и перейти к шагу 4.

Шаг 4. Вычислить такое значение $\lambda_i > 0$, что

$$f(x_i + \lambda_i h_i) = \min \{ f(x_i + \lambda_i h_i) \}.$$

Шаг 5. Положить $x_{i+1} = x_i + \lambda_i h_i$, положить $i = i + 1$ и перейти к шагу 2.

На шаге 3 производится поиск минимума по направлению (одномерный поиск). Поскольку он осуществляется численным алгоритмом, то, следовательно, значение минимума вычисляется с некоторой погрешностью. Исходя из этого, приведенный алгоритм можно считать принципиальным, а реализуемый алгоритм требует повторения поиска, который начинается из точки предшествующего минимума.

Различные варианты метода сопряженных градиентов отличаются только способом вычисления векторов h_i , по которым производится поиск минимума функции.

Рассмотрим процесс биортогонализации. Пусть дана симметрическая положительно определенная матрица размера $n \times n$. Построим две последовательности векторов $g_0, g_1, \dots$ и $h_0, h_1, \dots$ в R^n , таких, что

$$\langle g_i, g_j \rangle = 0 \quad \text{для всех } i \neq j$$

и

$$\langle h_i, Hh_j \rangle = 0 \quad \text{для всех } i \neq j.$$

Эти последовательности можно построить с помощью метода ортогонализации Грамма-Шмидта (прием «шнуровки»).

Пусть $g_0 \in R^n$ – произвольный вектор и $h_0 = g_0$. Далее последовательности векторов строятся следующим образом:

$$g_1 = g_0 - \lambda_0 Hh_0, \quad \lambda_0 = \frac{\langle g_0, g_0 \rangle}{\langle g_0, Hh_0 \rangle},$$

что дает $\langle g_0, g_1 \rangle = 0$;

$$h_1 = g_1 + \gamma_0 h_0, \quad \gamma_0 = -\frac{\langle Hh_0, g_1 \rangle}{\langle Hh_0, h_1 \rangle},$$

что дает $\langle h_0, Hh_1 \rangle = 0$.

Последующее построение

$$g_2 = \alpha g_0 + g_1 - \lambda_1 Hh_1, \quad h_2 = g_2 + \gamma_1 h_1 + \beta h_0.$$

Параметры λ_1 и α выбираются таким образом, чтобы

$$\langle g_0, g_1 \rangle = \langle g_1, g_2 \rangle = 0,$$

а γ_1 и β выбираются таким образом, чтобы

$$\langle h_0, Hh_2 \rangle = \langle h_1, Hh_2 \rangle = 0.$$

Построение продолжается до некоторого индекса $m \leq n$, при котором $g_m = h_m = 0$.

Используя этот алгоритм построения двух последовательностей векторов

$$g_{i+1} = g_i - \lambda_i Hh_i, \quad h_{i+1} = g_{i+1} + \gamma_i h_i,$$

где $h_0 = g_0$, а λ_i и γ_i выбраны таким образом, что

$$\langle g_{i+1}, g_j \rangle = 0 \quad \text{и} \quad \langle h_{i+1}, Hh_j \rangle = 0,$$

т.е.

$$\lambda_i = \frac{\langle g_i, g_i \rangle}{\langle g_i, Hh_i \rangle}, \quad \gamma_i = -\frac{\langle Hh_i, g_{i+1} \rangle}{\langle Hh_i, h_i \rangle}, \quad (4.63)$$

если знаменатели отличны от нуля и $\lambda_i = 0$, $\gamma_i = 0$ в противном случае. Можно показать, что все коэффициенты типа α , β и т.д. равны нулю. Векторы $g_0, g_1, \dots, g_m$ попарно ортогональны, а так как все они отличны от нуля, их общее число не может превосходить n , т.е. $m \leq n$.

Алгоритм Флетчера-Ривса.

Шаг 1. Выбрать точку $x_0 \in R^n$. Если $\nabla f(x_0) = 0$, то остановиться.

Шаг 2. Положить $i = 0$, $g_0 = h_0 = -\nabla f(x_0)$.

Шаг 3. Вычислить такое значение $\lambda_i > 0$, что

$$f(x_i + \lambda_i h_i) = \min \{ f(x_i + \lambda_i h_i) \}, \quad \lambda \geq 0.$$

Шаг 4. Положить $x_{i+1} = x_i + \lambda_i h_i$.

Шаг 5. Вычислить $\nabla f(x_{i+1})$.

Шаг 6. Если $\nabla f(x_{i+1}) = 0$, то остановиться. Иначе положить

$$g_{i+1} = -\nabla f(x_{i+1}), \quad h_{i+1} = g_{i+1} + \gamma_i h_i,$$

где $\gamma_i = \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle}$, положить $i = i + 1$ и перейти к шагу 3.

На шаге 2 вычисляется градиент целевой функции в стартовой точке. На шагах 3 и 4 определяется локальный минимум по выбранному направлению. Далее проверяются условия окончания поиска. Этот метод более эффективен, чем методы, в которых направление поиска задается заранее (например, метод наискорейшего спуска). Но сам алгоритм поиска является более сложным и требует разработки более сложных программ. В данном методе реализуется свойство градиентных методов, позволяющее работать с целевой функцией, имеющей разрывы производной (рис. 4.23).

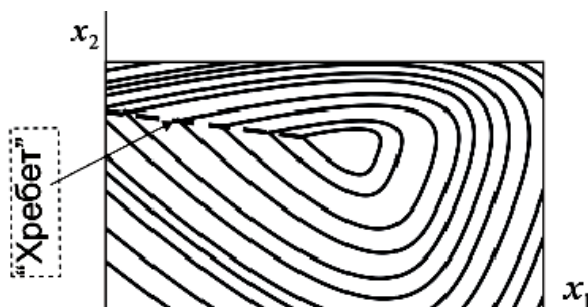


Рис. 4.23. «Хребет» – линия разрыва производной

Алгоритм Дэвидона-Флетчера-Пауэла (с переменной метрикой). Метод является наиболее широко используемым градиентным методом, обладающим хорошей устойчивостью. Относительным недостатком является необходимость хранения матрицы H_i ,

которая в задачах с большим количеством переменных может иметь большие размеры.

Шаг 1. Выбрать точку $x_0 \in R^n$. Если $\nabla f(x_0) = 0$, то остановиться; иначе перейти к шагу 2.

Шаг 2. Положить $i = 0$ и $H_0 = I$ (I – единичная матрица размерности $n \times n$, но можно взять любую симметрическую положительно определенную матрицу), положить

$$g_0 = \nabla f(x_0).$$

Шаг 3. Положить $h_i = -H_i g_i$.

Примечание. Матрица H_i называется *метрикой*, а метод – *методом с переменной метрикой*, поскольку матрица H_i изменяется на каждой итерации.

Шаг 4. Вычислить такое значение $\lambda_i > 0$, что

$$f(x_i + \lambda_i h_i) = \min \{ f(x_i + \lambda_i h_i) \}, \quad \lambda \geq 0.$$

Шаг 5. Вычислить $\nabla f(x_i + \lambda_i h_i)$.

Шаг 6. Если $\nabla f(x_i + \lambda_i h_i) = 0$, то остановиться, иначе положить

$$x_{i+1} = x_i + \lambda_i h_i,$$

$$g_{i+1} = \nabla f(x_{i+1}),$$

$$\Delta g_i = g_{i+1} - g_i,$$

$$\Delta x_i = x_{i+1} - x_i.$$

Вычислить

$$H_{i+1} = H_i - \frac{1}{\langle \Delta g_i, H_i \Delta g_i \rangle} \Delta x_i \langle H_i \Delta g_i + \frac{1}{\langle \Delta x_i, \Delta g_i \rangle} \Delta x_i \rangle \langle \Delta x_i$$

и перейти к шагу 3. (Матрица H_{i+1} , построенная таким образом, является симметрической положительно определенной, т.е. соответствует построению алгоритма поиска минимума функции методом сопряженных градиентов.)

Шаг 7. Положить $i = i + 1$ и перейти к шагу 3.

Примечание. Запись $\cdot \rangle \langle$ означает *диаду*. Если из двух векторов a и b составить, например, матрицу 3-го порядка, у которой (i, j) -й элемент матрицы равен $a_i b_j$:

$$\begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix}.$$

Такая матрица является диадой.

4.4. Методы случайного поиска (стохастическая аппроксимация)

Оригинальный подход, позволяющий обойти трудности применения детерминированных методов в случае многомерного пространства, предложен Бруксом и основан на случайном поиске. Пусть пространство проектирования представляет собой куб или гиперкуб со стороной, равной единице, и разделено на кубические ячейки путем деления на десять равных частей каждой стороны куба, соответствующей одному из проектных параметров. При размерности пространства поиска $N = 2$ число ячеек равно 100, при $N = 3$ – оно равно 1000. В общем случае при N -мерном пространстве число ячеек равно 10^N . Вероятность того, что выбранная наугад ячейка войдет в число 10 % наиболее перспективных ячеек равна 0,1. Так как при $N = 1$ нас будет интересовать одна ячейка из десяти, при $N = 2$ – одна из десяти лучших при общем количестве ячеек 100 и т.д. Вероятность того, что мы пропустим одну из 10 % наиболее перспективных ячеек, составит 0,9. Если случайным образом выбрать две ячейки, то вероятность пропуска будет 0,81. Вообще вероятность нахождения, по крайней мере, одной ячейки из наиболее перспективных, доля которых равна f , после N попыток составит

$$P = 1 - (1 - f)^N.$$

В табл. 4.4 указано, сколько ячеек надо выбрать случайным образом, чтобы обеспечить заданную вероятность при заданной доле наиболее перспективных ячеек. Из нее видно, что при случайной выборке 44 ячеек вероятность достижения $f = 0,1$ составит 99 %. Это очень неплохо, если вспомнить, что для 100 %-ного обеспече-

ния целевую функцию в случае пяти переменных пришлось бы вычислить 2 476 099 раз.

Таблица 4.4

f	Вероятность			
	0,80	0,90	0,95	0,99
0,1	16	22	29	44
0,05	32	25	59	90
0,01	161	230	299	459
0,005	322	460	598	919

Метод случайного поиска имеет два преимущества. Во-первых, он пригоден для любой целевой функции независимо от того, является она унимодальной или нет. Во-вторых, вероятность успеха при попытках не зависит от размерности рассматриваемого пространства. Хотя этот метод не позволяет непосредственно найти оптимальное решение, он создает подходящие предпосылки для применения в дальнейшем других методов поиска. Поэтому его часто применяют в сочетании с одним или несколькими методами других типов.

4.5. Поиск глобального минимума целевой функции

4.5.1. Поиск глобального минимума функции одной переменной (метод ломаных, метод покрытий)

Метод ломаных. Метод [1] применим к функциям, удовлетворяющим условию Липшица. Функция $f(x)$, заданная на отрезке $[a, b]$, удовлетворяет условию Липшица, если существует такая постоянная $L > 0$, что

$$|f(x_1) - f(x_2)| \leq L|x_1 - x_2| \quad \forall x_1, x_2 \in [a, b], \quad (4.64)$$

где L – постоянная Липшица.

Это условие геометрически отражает положение, что на любом участке рассматриваемой функции угол наклона хорды, соединяющей точки $(f(x_1), x_1)$ и $(f(x_2), x_2)$, не превышает значения постоянной L .

Процесс построения ломаной функции (рис. 4.24) начинается с выбора произвольной начальной точки x_0 и вычисления функции

$f(x_0)$ – точка C_0 . Из точки C_0 построим два отрезка $C_0 A_0$ и $C_0 B_0$ с угловыми наклонами L и $-L$ соответственно. Ломаная линия $A_0 C_0 B_0$ – первое приближение целевой функции. Следующая точка x_1 выбирается из условия наименьшего значения ординаты точки m_0 – пересечение линий $C_0 B_0$ и $C_1 A_1$. Совершенно очевидно, минимальное значение ордината точки m_0 примет лишь при условии, когда абсцисса точки C_1 совпадает с правой границей интервала $[a, b]$. (При другом выборе начальной точки x_0 , точка x_1 будет определяться левой границей интервала $[a, b]$.)

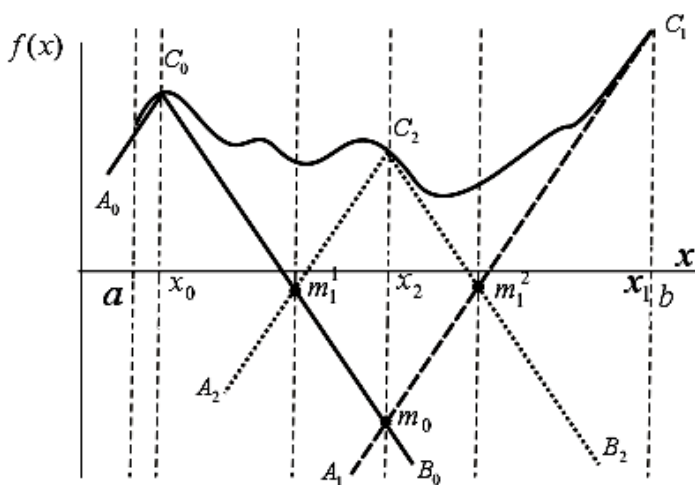


Рис. 4.24. Построение ломаной функции из условия Липшица.
Сплошная кривая – целевая функция

Ломаная линия $A_0 C_0 m_0 C_1$ является вторым приближением целевой функции. Абсцисса точки m_0 имеет значение x_2 . Определим точку C_2 , принадлежащую целевой функции, и из этой точки построим два отрезка $C_2 A_2$ и $C_2 B_2$ с угловыми наклонами L и $-L$ соответственно. Их пересечение с $A_0 C_0 m_0 C_1$ определит следующие точки m_1^1 и m_1^2 . Далее процесс построения продолжается по-

добным образом. В результате получаем приближение целевой функции снизу в виде ломаной линии (рис. 4.25).

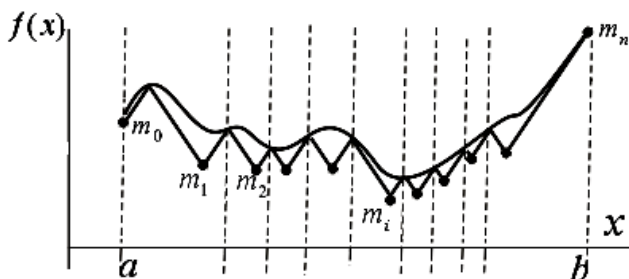


Рис. 4.25. Результирующая ломаная функция

Последовательность нижних точек результирующей ломаной функции $m_0, m_1, m_2, \dots, m_n$ (индексация переопределена с левого края интервала) позволяет выделить подынтервал, содержащий глобальный минимум. Достоинством этого метода является то, что он сходится при любом выборе начальной точки. Недостаток метода – необходимость заранее знать значение предельного угла наклона хорды L .

Метод покрытий. Существует целый ряд алгоритмов, в которых определенным образом строится последовательность отрезков $[a_i, b_i]$, полностью покрывающих интервал поиска $[a_{\text{нач}}, b_{\text{нач}}]$.

В точках a_i и b_i вычисляются значения целевой функции. Эти методы называются *методами покрытия* [1]. Самый простой алгоритм – *метод равномерного перебора точек* $x_i = a_{\text{нач}} + id$, где $d = x_{i+1} - x_i = \text{const}$.

Поиск глобального минимума целевой функции, удовлетворяющей условию Липшица (4.64), осуществляется методом сканирования интервала $[a_{\text{нач}}, b_{\text{нач}}]$ с достаточно большим шагом d по переменной x (рис. 4.26). Выделяются области $[a_0, b_0]$, в которых содержатся минимумы функции. Условия присутствия минимума функции на этом подынтервале определяются как $f(x_1) > f(x_2)$ и $f(x_3) > f(x_2)$. Каждая из этих областей исследуется на собствен-

ный локальный минимум (рис. 4.27). Среди локальных минимумов определяется глобальный минимум.

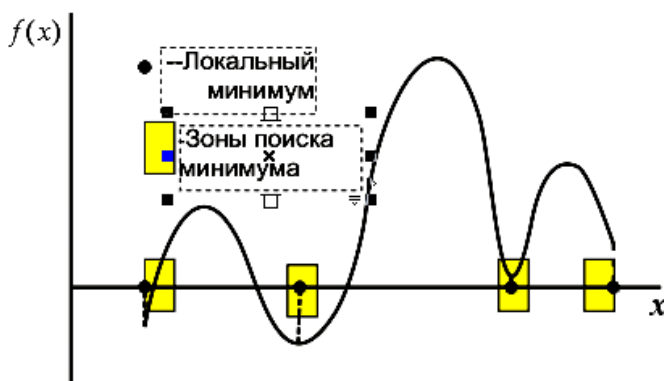


Рис. 4.26. Определение областей локальных минимумов

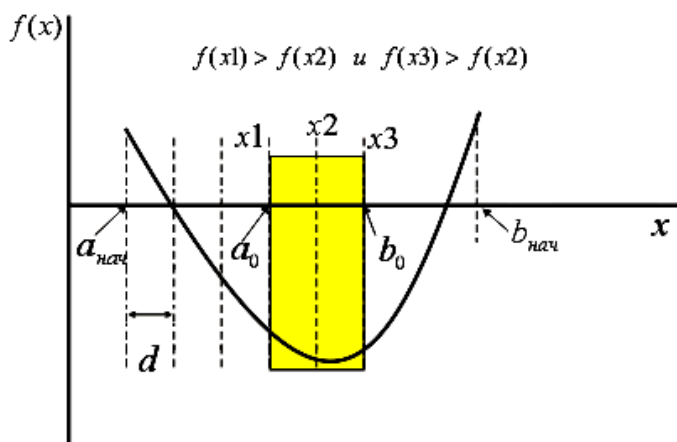


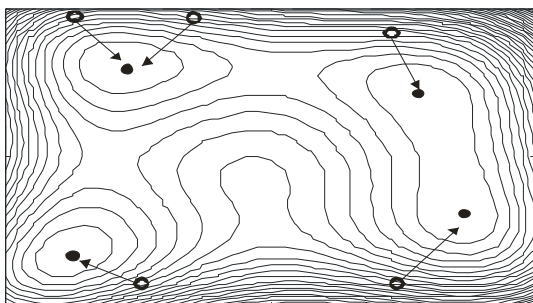
Рис. 4.27. Определение границ интервала $[a, b]$, содержащего локальный минимум, d – шаг сканирования

4.5.2. Поиск глобального минимума функции многих переменных

Целевая функция в общем случае может иметь сложную конфигурацию поверхностей (линий) уровня, которую при отсутствии

достаточно полной априорной информации, даже трудно себе представить. Поэтому в решении реальных задач приходится проводить численный эксперимент для определения возможных локальных минимумов. Например, функция Химмельблау имеет четыре минимума (рис. 4.28):

$$f(x, y) = (x \cdot x + y - 11) \cdot (x \cdot x + y - 11) + (x + y \cdot y - 7) \cdot (x + y \cdot y - 7).$$



- -Точка локального минимума
- -Стартовая точка

Рис. 4.28. Функция Химмельблау

При наличии многих экстремумов нужно построить циклический алгоритм с разными стартовыми точками. В результате алгоритм поиска (в зависимости от координат стартовой точки) будет тяготеть к различным минимумам. Если в результате достаточного числа поисков новых минимумов не обнаружится, то можно с большой вероятностью считать, что все локальные минимумы найдены. Определить глобальный минимум из ограниченного числа локальных минимумов не составит никакого труда.

При использовании алгоритмов, в которых производится вычисление производных, получить их аналитические выражения часто оказывается достаточно сложно. Применяемые конечно-разностные схемы вычисления производных должны быть хорошо апробированы и адаптированы к решаемой задаче. Сложности могут возникнуть при движении в направлении антиградиента в точках, близких к минимуму, так как в этой области градиент целевой функции близок к нулю. Процедуры аппроксимации целевой

функции при поиске одномерного минимума по выбранному направлению также вносят погрешности в принципиальный алгоритм поиска глобального минимума. Точность определения одномерного минимума определяется точностью используемых численных методов. В результате глобальный минимум оказывается недостижимым и требуется восстановление алгоритма, т.е. повторение поиска, для которого стартовой точкой является результат предшествующей итерации.

Задачи нелинейного программирования во многом присущи проблемам, возникающим при создании различных физических устройств или установок, а также исследовании сложных физических процессов. Целевая функция формируется на основе энергетических и динамических характеристик рассматриваемой системы. По своей форме целевая функция может иметь, например, следующий вид:

линейная модель –

$$f(x) = \sum_{i=1}^n a_i (x_i - x_i^*); \quad (4.65)$$

квадратичные модели –

$$f(x) = \sum_{i=1}^n a_i (x_i - x_i^*)^2, \quad f(x) = F \left(\sum_{i=1}^n a_i (x_i - x_i^*) \right)^2, \\ f(x) = \sum_{i=1}^n a_i (\varphi_i(x) - \varphi_i^*)^2, \quad f(x) = \sum_{i=1}^n a_i \left(\frac{(\varphi_i(x) - \varphi_i^*)}{\varphi_i^{\text{норм}}} \right)^2, \quad (4.66)$$

где $x = (x_1, x_2, \dots, x_n)$ – варьируемые параметры системы; a_i – весовые коэффициенты; x^* – параметр цели; $\varphi_i(x)$ – функция от варьируемых параметров системы; φ^* – функция параметра цели; $\varphi_i^{\text{норм}}$ – нормирующий коэффициент.

Используя весовые и нормирующие коэффициенты, можно произвести необходимое масштабирование переменных и целевой функции, а также выделить приоритетные параметры оптимизации.

В реальных физических задачах возможны разрывы целевой функции, которые заранее невозможно предугадать. Например, при

расчете динамики заряженных частиц в группирователе линейного ускорителя электронов может реализоваться такое распределение ускоряющего поля, при котором частицы не захватываются в режим ускорения, и, следовательно, нельзя сформировать целевую функцию. Решение подобных задач – особая область методов оптимизации [12].

4.6. Методы оптимизации в задачах с ограничениями

Задача формулируется как нахождение минимума целевой функции $f(x)$, определенной на множестве X :

$$\min \{f(x) : x \in X\} \quad (4.67)$$

при наличии ограничений

$$r_i(x) = 0 \quad \text{и} \quad g_i(x) \leq 0, \quad (4.68)$$

где $r_i(x)$, $g_i(x)$ – функции ограничения, определенные на множестве X .

Задачи условной оптимизации, как уже отмечалось, решаются сведением их к задачам безусловной оптимизации. Эта операция выполняется с учетом прямых и функциональных ограничений. Устранение прямых ограничений при переходе к безусловной оптимизации осуществляется соответствующим нормированием управляемых параметров. При прямом ограничении $a < u_i < b$ нормирование можно выполнить, например, следующим образом:

$$x_i = \operatorname{tg} \left[\pi \frac{u_i - (b+a)/2}{b-a} \right],$$

преобразовав ограниченный параметр u_i в неограниченный x_i . Функциональные ограничения устраняются путем конструирования обобщенной функции оптимизации с учетом типа ограничений.

Для решения подобных задач используются два различных по своей сути метода – *метод внешних штрафных функций* и *метод внутренних (барьерных) штрафных функций*.

Задачу нелинейного программирования с ограничениями можно записать в компактном виде, как

$$\min \{ f(x) : x \in C \subset R^n \}, \quad (4.69)$$

где функция $f(x)$ непрерывно дифференцируема; C – замкнутое множество в пространстве R^n .

Основная идея методов штрафных функций состоит в том, что от решения задачи (4.69) переходим к решению задачи безусловной оптимизации

$$\min \left\{ f(x) + \sum_{i=0}^n p_i(x) : x \in R^n \right\}, \quad i = 0, 1, \dots, n, \quad (4.70)$$

где $p_i(x)$ – штрафные функции.

В методе внешних штрафных функций штрафы $p_i(x)$ выбираются так, что точки вне C по мере удаления от границы становятся все более «невыгодными», при этом внутри C выполняются условия: либо $p_i(x) = 0$, либо $p_i(x) \rightarrow 0$. На рис. 4.29 приведены семейства внешних штрафных функций.

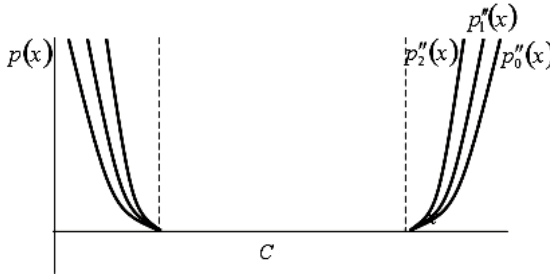


Рис. 4.29. Внешние штрафные функции

Последовательность непрерывных функций $p'_i(x)$, определенная на замкнутом множестве C , называется *последовательностью внешних штрафных функций*, если:

$$\begin{aligned} p'_i(x) &= 0 \quad \text{для всех } x \in C, \quad i = 0, 1, \dots, \\ p'_i(x) &> 0 \quad \text{для всех } x \notin C, \quad i = 0, 1, \dots, \\ p'_{i+1}(x) &> p'_i(x) \quad \text{для всех } x \notin C, \quad i = 0, 1, \dots, \\ p'_i(x) &\rightarrow \infty \quad \text{для всех } x \in C, \quad i \rightarrow \infty. \end{aligned} \quad (4.71)$$

В методе внутренних штрафных функций штрафы $p_i(x)$ выбираются так, чтобы точки, оптимальные для задачи (4.60), принадлежали внутренности C^0 множества C и $f(x) + p_i(x) \rightarrow f(x)$ для всех x из C^0 , а по мере приближения к изнутри – границе множества C

$$f(x) + p_i(x) \rightarrow \infty.$$

Последовательность непрерывных функций $p_i''(x)$, определенная на замкнутом множестве C , называется *последовательностью внутренних штрафных функций* (рис. 4.30), если:

$$0 < p_{i+1}''(x) < p_i''(x) \quad \text{для всех } x \in C, \quad i = 0, 1, \dots;$$

$$p_i''(x) \rightarrow 0 \quad \text{для всех } x \in C, \quad i \rightarrow \infty;$$

$$p_i''(x) \rightarrow \infty, \quad i = 0, 1, \dots, \quad \text{для всех } x, \text{ стремящихся к границам множества } C.$$

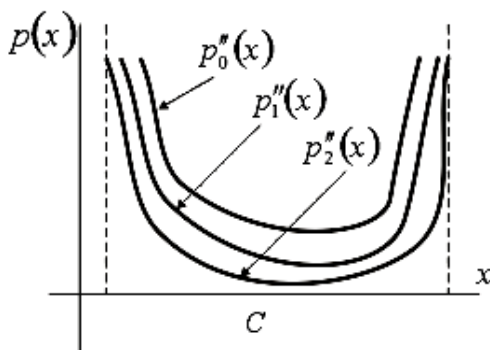


Рис. 4.30. Внутренние штрафные функции

Исходя из этих положений, в процессе поиска происходит как бы «отталкивание» внутрь от границ области поиска. Стартовая точка поиска должна выбираться из множества C^0 .

В реальных алгоритмах используются и смешенные технологии внешних и внутренних штрафных функций. Использование методов штрафных функций позволяет преобразовать задачу условной оптимизации в задачу безусловной оптимизации. При этом предпо-

лагается, что эта новая задача решается методами безусловной оптимизации.

Кроме методов штрафных функций существуют и другие методы решения задач условной оптимизации – методы центров, методы возможных направлений, методы проекций градиента и прямые методы, которые также будут рассмотрены в дальнейшем.

Контрольные вопросы

1. Дайте определение целевой функции.
2. Назовите методы поиска минимума одномерных унимодальных функций.
3. Перечислите методы исключения интервалов. Что такое дихотомия?
4. Что такое деление отрезка пополам?
5. Что такое золотое сечение?
6. Расскажите о методе Фибоначчи.
7. Перечислите сравнительные характеристики методов поиска минимума одномерных унимодальных функций.
8. Что такое полиномиальная аппроксимация? В чем суть метода с использованием квадратичной аппроксимации?
9. Расскажите о методах с использованием производных.
10. Расскажите о методе поиска с использованием кубической аппроксимации.
11. Перечислите сравнительные характеристики методов полиномиальной аппроксимации.
12. Дайте определение понятия минимума функций многих переменных. Расскажите о классификации методов.
13. Перечислите методы прямого поиска минимума функций многих переменных. В чем суть симплекс-метода?
14. Расскажите о методах спуска.
15. Расскажите о методе покоординатного спуска.
16. Расскажите о методе Хука-Дживса.
17. В чем суть метода сопряженных направлений Пауэла?
18. Перечислите градиентные методы. Расскажите о методе Коши.
19. Расскажите о методе Ньютона.

20. Перечислите методы сопряженных градиентов. Приведите алгоритм метода сопряженных градиентов (прототип).
21. Приведите алгоритм Флетчера-Ривса.
22. Приведите алгоритм Дэвидона-Флетчера-Пауэла (с переменной метрикой).
23. Перечислите методы случайного поиска (стохастическая аппроксимация).
24. Расскажите о поиске глобального минимума функции одной переменной.
25. Расскажите о поиске глобального минимума целевой функции. В чем суть метода ломаных?
26. Расскажите о методе покрытий.
27. Расскажите о поиске глобального минимума функции многих переменных.
28. Перечислите методы оптимизации в задачах с ограничениями.

СПИСОК ЛИТЕРАТУРЫ

1. Петров А.В., Черненко В.М. Проблемы и принципы создания САПР. Первая книга. М.: Высшая школа, 1990.
2. Математика в САПР. Первая книга. М.: Мир, 1988.
3. Козлов В.Н. Системный анализ, организация принятия решений. Учебное пособие. М., 2010.
4. Данчул А.Н., Полуян Л.Я. Системотехнические задачи создания САПР. М.: Высшая школа, 1990.
5. Системный анализ и принятие решений (словарь-справочник) / Под ред. В.Н. Волковой и В.Н. Козлова. М.: Высшая школа, 2004.
6. Трудношин В.А., Пивоваров Н.В. Математические модели технических объектов. М.: Высшая школа, 1988.
7. Вентцель Е.С. Теория вероятности. М.: Высшая школа, 2006.
8. Бусленко Н.П., Шрейдер Ю.А. Метод статистических испытаний. М., 1962.
9. Кузин Л.Т. Основы кибернетики. Учебное пособие. М.: МИФИ, 1970.
10. Васильев Ф.П. Численные методы решения экстремальных задач. М.: Наука, 1988.
11. Карманов В.Г. Математическое программирование. М.: Наука, 1980.
12. Реклейтис Г., Рейвиндран А., Рэгсдел К. Оптимизация в технике. В 2-х кн. М.: Мир, 1986.
13. Шуп Т. Решение инженерных задач на ЭВМ. М.: Мир, 1982.
14. Деннис Дж., Шнабель Р. Численные методы безусловной оптимизации и решение нелинейных уравнений. М.: Мир, 1988.
15. Полак Э. Численные методы оптимизации. М.: Мир, 1974.
16. Численные методы условной оптимизации / Под ред. Ф. Гилл и У. Мюррэй. М.: Мир, 1977.

17. Зангвилл У.И. Нелинейное программирование. М.: Советское радио, 1973.
18. Сеа Ж. Оптимизация. Теория и алгоритмы. М.: Мир, 1973.
19. Растрингин Л.А. Статистические методы поиска. М.: Наука, 1968.
20. Катковник В.Я. Линейные оценки и стохастические задачи оптимизации. М.: Наука, 1976.
21. Батухтин В.Д., Майборода Л.А. Оптимизация разрывных функций. М.: Наука, 1984.

САПР в электрофизике

Часть 1

Основы автоматизации проектирования

Редактор М.В. Макарова

Подписано в печать 15.11.2011. Формат 60х84 1/16

Уч.-изд. л. 11,0. Печ. л. 11,0. Тираж 200 экз.

Изд. № 4/4. Заказ № 78.

Национальный исследовательский ядерный университет «МИФИ».
115409, Москва, Каширское ш., 31.

ООО «Полиграфический комплекс «Курчатовский».
144000, Московская область, г. Электросталь, ул. Красная, д. 42

