



Р. П. ФЕДОРЕНКО

ПРИБЛИЖЕННОЕ РЕШЕНИЕ ЗАДАЧ
ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

СМБ

СМБ

Р. П. ФЕДОРЕНКО

ПРИБЛИЖЕННОЕ РЕШЕНИЕ ЗАДАЧ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

ИЗДАТЕЛЬСТВО «НАУКА»



Главная редакция
физико-математической
литературы



СПРАВОЧНАЯ
МАТЕМАТИЧЕСКАЯ
БИБЛИОТЕКА

Р. П. ФЕДОРЕНКО

**ПРИБЛИЖЕННОЕ
РЕШЕНИЕ ЗАДАЧ
ОПТИМАЛЬНОГО
УПРАВЛЕНИЯ**

**МОСКВА «НАУКА»
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1978**

22.18
Ф 33
УДК 519.6

Приближенное решение задач оптимального управления.
Р. П. Ф е д о р е н к о. Главная редакция физико-математической литературы, М., Наука, 1978. 488 с.

Книга посвящена методам приближенного решения задач оптимального управления в достаточно полном объеме: от теоретических выкладок до анализа выданных ЭВМ таблиц. Излагается теоретический материал, в основном связанный с важной в расчетах техникой вычисления функциональных производных. Описаны основные конструкции алгоритмов приближенного решения, использующие прямое решение уравнений принципа максимума, вариации в фазовом пространстве и вариации в пространстве управлений. Многочисленные примеры реализации алгоритмов для решения прикладных задач используются для иллюстрации характерных трудностей, методов их анализа, роли различных вычислительных приемов, обеспечивающих эффективность алгоритмов и надежность приближенных решений.

Книга предназначена научным работникам, занимающимся фактическим решением прикладных задач оптимизации.

ОГЛАВЛЕНИЕ

Предисловие	7
Введение	11
Г л а в а I. Элементы математической теории оптимального управления	16
§ 1. Общие замечания к первой главе	16
§ 2. Постановка вариационной задачи	21
§ 3. Дифференцирование функционалов, определенных на траекториях управляемой системы	29
§ 4. Функционалы, дифференцируемые по направлениям в функциональном пространстве	34
§ 5. Принцип максимума Л. С. Понтрягина — необходимое условие оптимальности управления	42
§ 6. Принцип максимума. Конечные вариации управления на множестве малой меры	55
§ 7. Некоторые обобщения задачи оптимального управления	61
§ 8. Принцип максимума в задачах с фазовыми ограничениями	75
§ 9. Принцип максимума — достаточное условие стационарности траектории	79
§ 10. Вопросы существования решений	81
§ 11. Вариационные задачи для ядерного реактора	96
§ 12. Задачи с уравнениями в частных производных	102
Г л а в а II. Методы приближенного решения задач оптимального управления	108
§ 13. Общие замечания к второй главе	108
§ 14. Методы решения краевой задачи для П-системы	114
§ 15. Метод вариаций в фазовом пространстве	120
§ 16. Метод вариаций в фазовом пространстве. Вычислительные схемы	127
§ 17. ϵ -метод Балакришнана	136
§ 18. Метод проекции градиента	140
§ 19. Метод последовательной линеаризации	164
§ 20. Метод последовательной линеаризации. Вычислительная технология	173
§ 21. Метод последовательной линеаризации. Задачи с функционалами, дифференцируемыми по Гато	180
§ 22. Метод поворота опорной гиперплоскости	188
§ 23. Приближенное решение задач со скользящим режимом	196
§ 24. Градиентный метод второго порядка	201
Г л а в а III. Решение задач	210
§ 25. Общие замечания к третьей главе	210
§ 26. Задача о брахистохроне	217

§ 27. Линейная задача быстрогодействия	227
§ 28. Задача о вертикальном подъеме ракеты-зонда. Нелинейная П-система	233
§ 29. Задача о вертикальном подъеме ракеты	238
§ 30. Задача о плоском движении тела переменной массы	249
§ 31. Оптимизация химического реактора	255
§ 32. Оптимизация производственного цикла	263
§ 33. Выбор оптимальных композиций защиты от излучения . . .	268
§ 34. Задача о стабилизации спутника	275
§ 35. Модельная задача с фазовым ограничением и разрывом фазовой траектории	289
§ 36. Оптимальный режим остановки реактора	295
§ 37. Задача о спуске космического аппарата	312
§ 38. Вариационные задачи, связанные с проектированием ядерного реактора	329
§ 39. Об одном способе аппроксимации не дифференцируемого функционала	338
§ 40. Некорректные задачи оптимального управления. Регуляризация численного решения	345
§ 41. Решение обратных задач математической физики. Вариационный подход	356
Г л а в а IV. Стандартные алгоритмы	369
§ 42. Основные свойства выпуклых множеств	369
§ 43. Метод Ньютона	377
§ 44. Дискретное динамическое программирование	386
§ 45. Поиск минимума. Гладкие задачи	389
§ 46. Поиск минимума. Негладкие задачи	407
§ 47. Линейное программирование. Симплекс-метод	417
§ 48. Линейное программирование. Итерационный метод	437
§ 49. Итерационный метод решения специальной задачи квадратического программирования	453
§ 50. Модифицированная функция Лагранжа	461
§ 51. Метод сопряженных градиентов	469
Литература	479
Предметный указатель	484
Указатель обозначений	487

ПРЕДИСЛОВИЕ

Основу этой книги составляет прежде всего опыт приближенного решения прикладных задач оптимального управления. Эта работа была начата автором в 1962 г. и продолжалась почти 15 лет. В течение этого времени задачи постепенно усложнялись, встречались трудности при их решении, надо было разбираться в причинах и вносить соответствующие изменения в метод решения, проводить многочисленные расчеты. Все это время автор следил за журнальной литературой по данной теме, пытался на основе имеющегося у него опыта оценить некоторые идеи, проводя, в частности, и вычислительные эксперименты. Таким образом, накопился достаточно большой материал. На его основе автором читались спецкурсы для студентов МФТИ и факультета прикладной математики и механики Воронежского университета, циклы обзорных лекций в 7-й зимней математической школе (г. Дрогобыч, 1974 г.) и во 2-й летней математической школе (г. Бендеры, 1977 г.). Однако при написании этой книги возникли значительные трудности.

Книга издается в серии СМБ и должна, в известной мере, служить справочником по различным методам решения задач оптимального управления. Однако в этом вопросе в настоящее время методов нет, если понимать метод как совокупность инструкций, следуя которым можно решить задачу данного типа. В этой книге описаны не методы, а скорее подходы к решению, разработанные вплоть до мелких деталей, многократно опробованные. Дело осложняется тем, что применение их к решению задач неизбежно связано с реализацией на ЭВМ, а это чрезвычайно остро ставит проблему объема вычислений (машинного времени). Поэтому метод, для которого доказана теорема о возможности получить решение с заданной точностью ценой конечного числа

операций (а это уже действительно метод), может оказаться совершенно не пригодным в качестве средства фактического решения прикладных задач, как в силу невыполнения предположений, принятых в доказательстве, так и в силу непосильного для современных ЭВМ (и ЭВМ обозримого будущего) объема вычислений. Такие примеры читатель найдет в этой книге.

Факт этот общеизвестен, и для нахождения минимума часто применяют алгоритмы, получившие название «эвристических». Этот термин, трактуемый иногда слишком широко, таит в себе опасность серьезного снижения требований к уровню вычислительной работы. Есть и другая опасность — предъявление к методам решения сложных задач требований, принятых в современной математике. Они не могут быть удовлетворены, но это не значит, что вычислительная математика находится вне науки. Эта книга написана с позиций, находящихся между двумя указанными крайними точками зрения. Разумеется, такая позиция неоднозначна и индивидуальна, оправданием ее может служить только основанная на ней практика решения прикладных задач.

В книге читатель найдет описание достаточно широкого спектра различных методов (будем все-таки употреблять этот термин) решения задач оптимального управления и сопутствующего их применению набора вычислительных приемов. Этим выполняются справочные функции книги. Вместе с тем автор считал своим долгом указать на трудности, которые возникнут при использовании того или иного метода. Изложение некоторых методов сопровождается критикой, иногда достаточно резкой. Исключить эту часть изложения невозможно: не предупредив читателя о возможных трудностях фактического решения задачи, автор ввел бы его в заблуждение. Разумеется, в этой части книга в известной мере субъективна. Но и здесь читатель найдет прежде всего объективную информацию. Автор никогда не позволял себе голословной критики, всегда четко и определенно указываются недостатки обсуждаемого метода, и автор настаивает на объективности этой части критики. Субъективным является отношение к этим недостаткам: можно ли, тем не менее, считать метод эффективным средством решения задач? В конце концов читатель этот вопрос должен будет решать сам, точка зрения автора для него необязательна. Еще раз подчеркнем, что даже самая резкая критика

служит в книге поводом для достаточно подробного описания некоторых методов, и читатель может ограничиться только этой информацией.

Другая трудность, с которой встретился автор, состоит в изложении вопросов вычислительной технологии. Вопросы, в сущности, мелких, но требующих достаточно ответственного решения. Без этого даже хорошая общая идея может не сработать. Попытки поднять эти вопросы до уровня науки и изложить их соответствующим образом (как это делается, например, в недавно переведенной монографии Э. Полака «Численные методы оптимизации» М., «Мир», 1974), представляются автору спорными. В книге реализован другой путь: автор не пытался изложить технологию вычислений в самом общем и абстрактном виде, предпочитая показать, как решаются эти вопросы в конкретных задачах. При этом используются соображения здравого смысла. Поняв их в простом частном случае, читатель без труда сможет (если сочтет нужным) использовать аналогичные соображения в своей работе, соответствующим образом видоизменив их.

Большое внимание в книге уделено неудачным расчетам. Очень важно уметь обнаружить ошибочность расчета, понять и проанализировать причину неудачи. В этом случае такой неудачный расчет оказывается (с методической точки зрения) даже поучительнее удачного. Развитие метода всегда так или иначе связано с преодолением встретившихся трудностей. Здесь нужно только избежать самой большой опасности — принять ошибочный расчет за решение задачи. В вычислительной математике это — одна из самых серьезных неприятностей. Ведь контроль того или иного опубликованного результата редко может быть осуществлен традиционным в математике чисто логическим путем. Он требует проведения вычислений, а это связано с большими затратами чисто технической работы.

Как уже отмечалось, эта книга основана на опыте решения прикладных задач. Она не могла бы появиться без сотрудничества автора с коллективами инженеров-физиков, которым принадлежит постановка ряда оригинальных задач, подготовка необходимой информации для конкретных расчетов (часто весьма объемистой), содержательная интерпретация полученных решений. Их интерес к проводившимся автором расчетам был стимулом,

значение которого трудно переоценить. Автору приятно выразить искреннюю благодарность А. А. Абагяну, А. П. Дубинину, В. В. Орлову, А. П. Суворову (в связи с задачами § 33), В. Н. Артамкину (в связи с задачами § 36), А. Д. Климову, И. Л. Чихладзе (в связи с задачами § 32 и § 38), Л. П. Беркович и И. В. Отрощенко, принимавшим участие в решении задач § 36, а также многим другим.

Большое значение для автора имела работа в коллективе Института прикладной математики. Характерное для этого коллектива стремление найти интересные и новые области приложения вычислительных методов, установить творческий контакт с физиками, инженерами, химиками, медиками и представителями других естественнонаучных дисциплин и получить на ЭВМ интересные для этих ученых результаты во многом определило стиль и содержание этой книги. Наконец, автор считает своим долгом отметить исключительное влияние своих учителей М. В. Келдыша и И. М. Гельфанда. У них автор старался учиться тому, что такое математика вообще и вычислительная в частности.

Автор

ВВЕДЕНИЕ

Математическая теория оптимального управления начала особенно интенсивно развиваться после выхода в свет известной монографии Л. С. Понтрягина и его сотрудников [65]. Можно даже сказать, что эта теория стала модной. Этому, в частности, способствовал и тот факт, что задачи создания оптимальных конструкций, режимов управления и т. д. возникают в самых различных прикладных областях. Одновременно с чисто теоретическими исследованиями началась и разработка приближенных методов решения задач оптимального управления. Поток работ на эту тему велик и не ослабевает до настоящего времени. Предлагаемая читателю книга является попыткой подвести итоги этой работы, разобраться в том, что уже удалось сделать, а что — пока еще нет, каковы реальные успехи на этом пути. Следует предупредить читателя, что вычислительная математика обладает обманчивой внешней простотой, и создание вычислительных методов для решения тех или иных задач кажется зачастую очень бесхитростным занятием, а в то же время актуальность разработки эффективных методов вычислений постоянно подчеркивается. Дело в том, что понятие «эффективный вычислительный метод» после появления ЭВМ претерпело существенное изменение. В «домашинную» эру можно было говорить о создании эффективного метода решения какого-то класса задач, если была доказана теорема о том, что с любой заданной точностью задачу можно решить ценой конечного числа операций над конечным множеством чисел. Само же число операций особенно не обсуждалось: в любом случае оно было очень большим.

И сейчас продолжают исследования подобного рода, но их, в сущности, следует относить не к вычислительной математике, а, например, к функциональному анализу или к теории аппроксимации. В настоящее время, когда мощные ЭВМ стали доступны огромному числу научных работников, об эффективном методе решения можно говорить лишь в том случае, если действительно решаются прикладные задачи данного типа на реальных ЭВМ за реальное машинное время. К сожалению, для большинства используемых в практических расчетах методов нет эффективных

оценок, позволяющих по заданной точности расчета определить необходимые для его реализации ресурсы памяти и машинного времени. Поэтому оценка подобных методов осуществляется, как правило, на основании вычислительного опыта, а вычислительная математика оказывается наукой в известной мере экспериментальной. Это признается почти всеми, но соответствующие традиции освещения и истолкования экспериментального материала еще не сложились. Во многих работах можно встретить утверждения о том, что предлагаемый метод оказался надежным, дал хорошие результаты, показал высокую эффективность и т. д. Часто подобные утверждения не подкреплены публикацией данных, которые придали бы им хоть сколько-нибудь определенный смысл: читатель не получает информации ни о сложности решенных задач, ни об объеме вычислений, ни о качестве результатов, ни о возможности решения задачи другими, уже известными методами.

Сейчас создано очень много вычислительных методов, в частности, и для решения задач оптимального управления. Разумеется, они не решают проблемы полностью, но не любой формально новый метод является шагом вперед. Дальнейшее развитие вычислительных методов требует четкого представления о том, что уже сделано, а что еще не удастся, ради чего предпринимаются усилия при создании нового метода. Без этого велика вероятность появления лишь формально новых методов вычислений, которые не лучше (а часто и хуже) существующих там, где они работают, и не дают ничего в тех задачах, с которыми существующие методы не справляются. Этими замечаниями в значительной мере определяется характер настоящей книги. Ее основное содержание — методы приближенного решения задач оптимального управления. Автор ставил целью не только познакомить читателя с основными идеями конструкций вычислительных алгоритмов, но и с тем, как эти идеи доводятся до конца, до фактического решения задач, какие технические трудности приходится при этом преодолевать и как это делается. Речь идет о совокупности приемов, образующих, так сказать, вычислительную технологию. Это — очень важная часть практической вычислительной работы, без грамотного оформления которой никакую идею не удастся довести до успешного расчета. К сожалению, эта совокупность знаний и навыков еще не выросла (и едва ли когда-нибудь дорастет) до уровня науки. Эта технология и есть то, что обычно называют «здоровым смыслом», «вычислительным опытом» и т. д. Автор попытался познакомить читателя и с этой стороной вычислительной математики, разумеется, лишь в той мере, в какой он сам ее понимает. Теперь несколько замечаний о содержании книги, назначении ее отдельных частей и характере изложения. Весь материал естественно разбивается на четыре главы, посвя-

щенные относительно самостоятельным вопросам, объединенным общей целью — познакомить читателя с методами приближенного решения задач оптимального управления в достаточно полном объеме — начиная с чисто теоретических выкладок и кончая анализом выданного машиной числового материала.

Первая глава — «Элементы математической теории оптимального управления» (§§ 1—12) — содержит минимум необходимых теоретических результатов, без которых братья за численное решение задач оптимального управления нельзя. Хотя входящий в эту главу материал можно в той или иной форме найти в большом числе руководств, она представляется автору необходимой по следующим причинам:

1. В главу включены лишь те элементы общей теории, которые имеют прямое и непосредственное приложение в конструкциях численных методов и в практике фактического решения прикладных задач. Многие разделы теории, как бы ни были они изящны и глубоки (например, теория линейных задач оптимального управления), опущены, и с ними читатель может познакомиться по другим книгам. В принципе, читатель, совершенно незнакомый с математической теорией оптимального управления, усвоив лишь теоретический материал первой главы, сможет понять и все остальное.

2. В этой главе вводится система понятий, терминов, основных математических объектов и соответствующих им обозначений, которая используется в книге.

3. Изложение теории (в частности, доказательство принципа максимума) дается в редакции, отличающейся от общепринятой, но более подходящей для основного содержания книги. Большое внимание уделяется технике вычисления функциональных производных при различных способах определения функционалов. Эта техника сама по себе очень важна, особенно при численном решении задач. Кроме того, читатель, владеющий этой техникой, может, так сказать, сэкономить на теории. В современной литературе появилось много публикаций, где формулируется новый тип вариационной задачи и доказывается соответствующий вариант принципа максимума. В настоящей книге автор придерживается следующей точки зрения: подобные исследования отличаются друг от друга в основном лишь формой уравнения, связывающего управление и состояние объекта, и формой определения функционалов. Следствием этого является и различие в необходимых для нахождения функциональных производных вычислениях. Поэтому эту техническую часть следует выделить и изучить отдельно. Все остальное формально укладывается в некоторую общую схему (см. § 1).

Вторая глава — «Методы приближенного решения задач оптимального управления» (§§ 13—24). Каждый параграф этой главы

содержит описание одного из возможных подходов к построению метода приближенного решения задач оптимального управления. Их не так уж много, и это находится в видимом противоречии с обилием работ, претендующих на создание нового метода. Стоит разобраться в этом вопросе. В каждом методе приближенного решения задач оптимального управления можно достаточно четко выделить три слоя:

1. Класс задач, для которых предназначен метод. Например, это могут быть задачи для управляемых систем, описываемых обыкновенными уравнениями, уравнениями с запаздыванием, уравнениями с частными производными и т. д.

2. Общая идея конструкции численного метода.

3. И, наконец, элементы вычислительной технологии, возникающие при реализации метода на ЭВМ.

Итак, мы имеем большое число возможных типов вариационных задач, некоторое число основных идей численного их решения и достаточно большое число возможных технологических оформлений. И каждый из элементов этих трех уровней может сочетаться если и не с каждым, то с большим числом элементов соседнего уровня. Вот эта-то комбинаторика и создает (в значительной мере) видимое разнообразие методов приближенного решения. Однако в этих комбинациях могут содержаться и очень ценные предложения, когда есть достаточно веские основания утверждать, что для данного специального класса задач следует выбрать именно данный подход и дополнить его именно одним конкретным вариантом технологии, а при других комбинациях получатся заметно менее эффективные или трудно реализуемые методы. Этой трехслойной структуре проблемы приближенного решения задач оптимального управления и соответствуют первые три главы книги. Во второй главе каждый возможный подход описан достаточно подробно, но самый низший уровень — технология вычислений — естественно, не излагается: это уже материал третьей главы. Выше мы отмечали, что основных конструкций приближенных методов оказалось не так уж много. Автор надеется, что читатель, разобравшийся в этом материале, без труда убедится, например, в том, что очень большое число предложенных в разное время и в разных странах методов являются несущественными модификациями простейших вариантов метода проекции градиента.

Третья глава — «Решение задач» — содержит большое число примеров фактической реализации того или иного метода. Хотя большая часть решавшихся задач имеет конкретное прикладное значение, с этой «физической» точки зрения они не обсуждаются. Не обсуждается и прикладное значение полученных приближенных решений. Все эти задачи рассматриваются исключительно с методической точки зрения, наибольшее внимание уделяется

самому процессу получения приближенного решения, характерным трудностям и способам их преодоления. Выше подчеркивалось значение аккуратного подхода к вопросам техники вычислений. Попытка их выделения и изложения в абстрактной, общей форме автору не удалась: получалось неубедительно и голословно. Остался единственный путь: показать, как решаются эти вопросы в конкретных ситуациях, и каков эффект того или иного приема. Большое число примеров не случайно, так как в каждой задаче наиболее выпукло проявляется одна какая-то сторона вычислительной технологии. Кроме того, подробные комментарии к процессу решения многих задач преследуют еще одну цель: ввести читателя, если так можно сказать, в «кухню» вычислительной работы. Отсюда обилие графиков, таблиц, анализ результатов, выявление возможных ошибок, то или иное объяснение возникающих затруднений, попытки (удачные и неудачные) решения одной и той же задачи разными средствами и т. д. Без этого вычислительная работа немыслима, а передать другому весь этот опыт можно, видимо, только заставив в какой-то мере пройти тот же путь, который прошел автор. Кроме того, этот материал наполняет конкретным содержанием утверждения об эффективности метода, об успешном решении прикладных задач. Читатель может увидеть, что же в конце концов получается в расчетах, и сам, в меру своей требовательности, оценить результаты как удовлетворительные или нет, а не полагаться на субъективные оценки автора. Наконец, читатель, желающий внести свой вклад в развитие приближенных методов, может использовать многие задачи в качестве методических тестов и сравнить свои достижения с тем, что уже получено. Изложение некоторых, часто популярных и имеющих хорошую репутацию в литературе, вычислительных приемов сопровождается критическим комментарием. Разумеется, этот скептицизм является личным делом автора и читатель не обязан его разделять. Во всех подобных случаях приводятся доводы и соображения, на которых основана точка зрения автора, а часто и подтверждающий ее экспериментальный материал.

Четвертая глава — «Стандартные алгоритмы» — включает в себя §§ 42—51, каждый из которых посвящен тому или иному стандартному алгоритму. Эти алгоритмы объединены общим назначением — они используются в качестве рабочего инструмента при численном решении задач оптимизации. Действие этих алгоритмов также иллюстрируется числовыми примерами.

В книге принята сквозная нумерация параграфов. В каждом параграфе формулы нумеруются одним числом, при ссылке на формулу данного параграфа указывается номер формулы, при ссылке на формулу из другого параграфа — номер параграфа и формулы. Та же система нумерации принята и для определений, лемм и теорем.

ГЛАВА I

ЭЛЕМЕНТЫ МАТЕМАТИЧЕСКОЙ ТЕОРИИ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

§ 1. Общие замечания к первой главе

В этой главе излагается минимальный теоретический материал, необходимый и достаточный для понимания всего остального, составляющего основное содержание книги. Тем, кто знаком с математической теорией оптимального управления, полезно познакомиться с этой главой, чтобы привыкнуть к принятой в книге терминологии и системе обозначений. Впрочем, они не очень отличаются от тех, которые используются в ставшей уже классической монографии [65]. Читатель, не разбиравший подробно первых глав этой монографии и знакомый с теорией по упрощенным изложениям в руководствах сугубо прикладного направления (или совсем незнакомый с ней), должен основательно усвоить хотя бы содержание §§ 1—7; без этого трудно будет понять все остальное. Заметим, что хотя данная книга имеет явно прикладной характер, в изложении теоретического материала она гораздо ближе к чисто теоретическим работам типа [65], [34]. Это связано с существом дела. Читатель убедится, что математические тонкости доказательства принципа максимума, которые мы специально выделяем и подчеркиваем в §§ 5, 6, имеют самое прямое отношение к приближенному решению задач. Кстати, из многих известных сейчас схем доказательства принципа максимума (так же, как и других приведенных в книге теорем) автор специально отобрал не самые краткие, общие и изящные, но те, которые более или менее явно индуцируют методы приближенного решения.

Большая часть исследований, связанных с принципом максимума, проводится по следующей общей схеме. В ней в абстрактной форме отражены основные преобразования и рассуждения

Общая постановка задачи. Пусть определено некоторое замкнутое ограниченное множество U в функциональном пространстве; элементы этого пространства будем обозначать

и. Пусть определены функционалы от u :

$$F_0(u), F_1(u), \dots, F_m(u).$$

Задача состоит в определении u из условий

$$\begin{aligned} \min_u F_0(u), \\ F_i(u) = 0 (\leq 0), \quad i = 1, 2, \dots, m, \\ u \in U. \end{aligned} \quad (1)$$

Это есть достаточно общая постановка задачи математического программирования, частным случаем которой является задача оптимального управления. Для последней характерно следующее усложнение. Функционалы $F_i(u)$ задаются явными формулами, содержащими, кроме u , еще и аргумент x , являющийся точкой другого функционального пространства, причем u и x связаны операторным уравнением

$$R(x, u) = 0. \quad (2)$$

Оно предполагается разрешимым относительно x при заданном u . Таким образом, для $F_i(u)$ имеем формулы

$$F_i(u) = \Phi_i(x, u), \quad (3)$$

причем зависимости $\Phi_i(x, u)$ считаются явно заданными, в то время как $F_i(u)$ есть лишь абстрактное обозначение, выражающее принципиальную возможность вычислить F , зная u . Фактически эта возможность реализуется следующими вычислениями: задав u , нужно определить x из уравнения $R(x, u) = 0$, затем вычислить $\Phi_i(x, u)$, что и будет F_i . Формальная схема исследования некоторой точки u — предполагаемого решения задачи — состоит в анализе последствий малого возмущения δu . Пусть все функционалы $F_i(u)$ — дифференцируемы. Тогда следует выяснить, разрешима ли задача

$$\begin{aligned} \delta F_0(\delta u) &= \frac{\partial F_0}{\partial u} \delta u < 0, \\ \delta F_i(\delta u) &= \frac{\partial F_i}{\partial u} \delta u = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \\ u + \delta u &\in U. \end{aligned} \quad (4)$$

Если эта линеаризованная задача неразрешима, точка u удовлетворяет необходимому условию оптимальности. В противном случае в окрестности u есть «лучшая» точка, и многие методы приближенного решения задачи математического программирования основаны на следующей простой схеме: если задача для δu разрешима, следует ее решить, перейти к точке $u + \delta u$ и исследовать

ее таким же образом. Так получаем процесс построения минимизирующей последовательности точек u^0, u^1, \dots . Нужно только добавить два существенных технологических момента:

1. К условиям задачи (4) добавляется условие $\|\delta u\| \leq \varepsilon$, где ε — малое число; обеспечивающее правомерность использования линейного приближения, а первая строка (4) формулируется в виде $\min_{\delta u} \frac{\partial F_0}{\partial u} \delta u$.

2. В силу нелинейности задачи условия $\frac{\partial F_i}{\partial u} \delta u = 0$ не обеспечивают выполнения условий $F_i(u^k) = 0$; происходит накопление ошибок, имеющих порядок $O(\|\delta u\|^2)$. Для предотвращения этого вторая строка в (4) формулируется в виде

$$F_i(u) + \frac{\partial F_i}{\partial u} \delta u = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m.$$

Теперь обратимся к задачам оптимального управления (1), (2), (3). Сначала мы получаем формулы для δF в терминах δu и δx ,

$$\delta F = \frac{\partial F}{\partial u} \delta u = \frac{\partial \Phi}{\partial u} \delta u + \frac{\partial \Phi}{\partial x} \delta x, \quad (5)$$

причем слагаемое $\frac{\partial \Phi}{\partial x} \delta x$ следует заменить на функционал от δu . Это исключение достигается использованием уравнения в вариациях

$$R_x \delta x + R_u \delta u = 0, \quad (6)$$

и тождества Лагранжа

$$(\psi, R_x \delta x) = (R_x^* \psi, \delta x).$$

Теперь подберем ψ специальным образом, как решение уравнения $R_x^* \psi = -\frac{\partial \Phi}{\partial x}(x, u)$, а из уравнения в вариациях (6) выразим $R_x \delta x = -R_u \delta u$. Тогда

$$\frac{\partial \Phi}{\partial x} \delta x = -(R_x^* \psi, \delta x) = -(\psi, R_x \delta x) = (\psi, R_u \delta u) = (R_u^* \psi, \delta u).$$

Таким образом, получаем из (5) общую формулу для функциональной производной

$$\frac{\partial F(u)}{\partial u} = \frac{\partial \Phi}{\partial u} + R_u^* \psi. \quad (7)$$

Проведенная выше операция исключения δx из первичной формулы для δF (5) есть не что иное, как проектирование градиента $\left\{ \frac{\partial \Phi}{\partial x}, \frac{\partial \Phi}{\partial u} \right\}$ в произведении пространств $\{x\} \times \{u\}$ на линейное подпространство, касающееся в точке $\{x, u\}$ многообразия, выделяемого уравнением связи $R(x, u) = 0$. Это подпространство имеет и «явное

выражение» $\{x + \delta x, u + \delta u\}$, причем δx и δu связаны линейным уравнением в вариациях (6). Далее, условия

$$F_i(u) = 0 \text{ (или } \Phi_i(x, u) = 0, i = 1, 2, \dots, m)$$

также выделяют некоторое многообразие, а уравнения

$$\frac{\partial F_i}{\partial u} \delta u = 0 \text{ (или } \frac{\partial \Phi_i}{\partial x} \delta x + \frac{\partial \Phi_i}{\partial u} \delta u = 0), \quad i = 1, \dots, m$$

определяют касательное в точке u (или $\{x, u\}$) линейное подпространство $\{x + \delta x, u + \delta u\}$, и в вычислениях часто используется проекция градиента F_0 на это подпространство. Эту последнюю проекцию можно вычислить двумя, внешне разными, способами:

1. Сначала первичные градиенты $\left\{ \frac{\partial \Phi_i}{\partial x}, \frac{\partial \Phi_i}{\partial u} \right\}$ проектируются, и превращаются по (7) в $\frac{\partial F_i}{\partial u}$, а затем $\frac{\partial F_0}{\partial u}$ проектируется на подпространство, выделяемое условиями

$$\frac{\partial F_i(u)}{\partial u} \delta u = 0, \quad i = 1, 2, \dots, m.$$

2. Градиент $\left\{ \frac{\partial \Phi_0}{\partial x}, \frac{\partial \Phi_0}{\partial u} \right\}$ сразу проектируется на линейное подпространство, выделяемое условиями

$$R_x \delta x + R_u \delta u = 0; \quad \frac{\partial \Phi_i}{\partial x} \delta x + \frac{\partial \Phi_i}{\partial u} \delta u = 0, \quad i = 1, \dots, m.$$

Оба способа дают одно и то же, это есть аналог теоремы о трех перпендикулярах. Заметим, что выше все преобразования были проведены формально; мы не обсуждали, например, вопроса о том, в каком пространстве лежат элементы δu , δx (поскольку существенно использовалось скалярное произведение, то проще всего предположить пространство гильбертовым), о законности тех или иных преобразований, о разрешимости встречающихся уравнений. Тем не менее эта абстрактная схема полезна, и она составляет, так сказать, внешний каркас почти всех исследований, связанных с принципом максимума и приближенным решением экстремальных задач. Различные типы таких задач отличаются в основном конкретными формами уравнений связи $R(x, u) = 0$ и видом функций $\Phi_i(x, u)$. Эту сторону вопроса мы будем считать чисто технической и не оказывающей существенного влияния на выбор алгоритма численного решения задачи. Однако это замечание не следует толковать слишком широко: книга посвящена численному решению задач оптимального управления, а не более общей задаче математического программирования. Тем не менее с этим связано определенное ограничение на характер уравнения связи $R(x, u) = 0$. Под задачей оптималь-

ного управления мы будем понимать задачу, в которой $R(x, u) = 0$ есть дифференциальное уравнение для x (обыкновенное, в частных производных, уравнение с запаздыванием, интегродифференциальное), в коэффициенты (начальные данные, краевые условия и т. д.) входит «управление» u . Это обстоятельство, хотя и оставляет задачу достаточно неопределенной, все же выделяет некоторый класс, и предлагаемые в книге алгоритмы существенно используют его специфику. Отметим здесь же некоторые из конкретных свойств этого класса задач, имеющих серьезное значение при конструировании методов приближенного решения:

1. Обычно уравнение $R(x, u) = 0$ разрешимо относительно x при более или менее произвольном u , но при почти любом x неразрешимо относительно u .

2. В тех частных задачах, в которых, как, например, в задачах классического вариационного исчисления, x и u формально равноправны ($R(x, u) = 0$ разрешимо как относительно x , так и относительно u), градиент функционала F при выборе x в качестве независимого аргумента оказывается неограниченным (дифференциальным) оператором. Если же в качестве независимого аргумента выбирается u , градиент оказывается ограниченным.

3. Аргументы задачи x и u оказываются элементами разных функциональных пространств: u обычно бывает произвольной (измеримой) функцией, а x — сравнительно гладкой.

4. В задачу входит совокупность дополнительных условий, ограничивающих выбор u . Эти условия имеют вид

- 1) $u \in U$,
- 2) $F_i(u) = 0, \quad i = 1, 2, \dots, m,$
- 3) $x \in G.$

Формально можно было бы все эти условия объединить в единое условие $u \in U$, что сразу сделало бы теорию более компактной, общей и изящной. Однако в теории оптимального управления эти виды ограничений различаются и изучаются отдельно. В этой книге мы тоже придерживаемся такого подхода, и это связано с существом дела. Отнесение входящих в конкретную задачу ограничений к одной из трех форм производится по следующему содержательному признаку: для любой точки u проверка условия $u \in U$ очень проста и, что существеннее, операция проектирования любой точки v на U (т. е. решение задачи $\min_{u \in U} \|u - v\|$)

может считаться (сравнительно с остальными операциями) элементарной. Проверка условий $F_i(u) = 0, i = 1, 2, \dots, m$, и соответствующее проектирование требуют уже более сложных вычислений. Наконец, условие $x \in G$ (в такой форме задаются так называемые ограничения в фазовом пространстве) проверяется обычно

вычислениями, немногим более сложными, чем для условий $F_i(u)=0$, но соответствующая операция проектирования точки u становится очень сложной. Все это сказывается и на конструкциях алгоритмов приближенного решения задачи: каждый из трех видов ограничений требует своего подхода, и сложность их возрастает в соответствии со сложностью операции проектирования.

§ 2. Постановка вариационной задачи

1. Управляемые системы. Классическое вариационное исчисление возникло в связи с задачами следующего типа: найти функцию $x(t)$, $0 \leq t \leq T$, удовлетворяющую краевым условиям $x(0)=X_0$, $x(T)=X_1$ и минимизирующую значение функционала (здесь $\Phi(x, y)$ — заданная функция)

$$F[x(\cdot)] \equiv \int_0^T \Phi[x(t), \dot{x}(t)] dt. \quad (1)$$

Сразу же поясним обозначения, систематически применяющиеся в дальнейшем: $x(\cdot)$ будет обозначать функцию, взятую целиком, как элемент функционального пространства, а $x(t)$ — есть обозначение числа (или вектора, если $x(t)$ — вектор-функция), являющегося значением $x(\cdot)$ в точке t . Иногда употребление $x(t)$ в смысле $x(\cdot)$ не вносит путаницы, но в некоторых случаях их следует различать. Таким образом, левая часть (1) есть абстрактное обозначение функционала — числовой функции, аргументом которой являются точки функционального пространства, а правая часть определяет фактический способ его вычисления *). В классическом вариационном исчислении можно выделить следующие важные разделы:

У с т а н о в л е н и е д и ф ф е р е н ц и р у е м о с т и функционала и вычисление производной. Функцию $w(t)$ будем называть *функциональной производной* в смысле Фреше и обозначать

$$\frac{\partial F[x(\cdot)]}{\partial x(t)} = w(t), \quad (2)$$

если для всех достаточно малых возмущений $\delta x(\cdot)$ имеет место формула

$$F[x(\cdot) + \delta x(\cdot)] = F[x(\cdot)] + \int_0^T w(t) \delta x(t) dt + O(\|\delta x\|^2). \quad (3)$$

*) Иногда в (1) используется абстрактное обозначение $F[x(t)]$; это — неудачная символика, так как правая часть (1) не зависит от «немого» аргумента t .

Функционал $F[x(\cdot)]$ в этом случае называется *дифференцируемым* *); разумеется, производная вычисляется, в общем случае, лишь в данной точке $x(\cdot)$ функционального пространства. Классический результат Эйлера состоит в том, что для (1)

$$\frac{\partial F[x(\cdot)]}{\partial x(t)} \equiv -\frac{d}{dt} \Phi_{\dot{x}}[x(t), \dot{x}(t)] + \Phi_x[x(t), \dot{x}(t)]. \quad (4)$$

Приравняв правую часть (4) нулю, получаем необходимое условие, которому должна удовлетворять искомая функция $x(t)$, — известное уравнение Эйлера

$$-\frac{d}{dt} \Phi_{\dot{x}}(x, \dot{x}) + \Phi_x(x, \dot{x}) = 0. \quad (5)$$

Решив его с учетом краевых условий для $x(t)$, получим стационарную точку функционала $F[x(\cdot)]$; она может оказаться как точкой минимума (локального), так и точкой максимума или точкой «перегиба» (имеется в виду точка функционального пространства).

Анализ второй вариации функционала. Он позволяет более или менее эффективно выяснить, является ли исследуемое решение уравнения (5) точкой минимума $F[x(\cdot)]$ (локального, разумеется), или оно является стационарной точкой другого типа. Во многих прикладных задачах такие вопросы решаются установлением единственности решения уравнения Эйлера и ограниченности функционала снизу.

Анализ задачи. Важную часть теории вариационных задач составляет анализ задачи «в целом» и связанные с ним достаточные условия экстремума Вейерштрасса. Построенная им теория получила естественное обобщение в виде теории динамического программирования.

Теоретические вопросы, связанные с уточнением постановки задачи. Постановка задачи должна содержать четкое описание функционального пространства, на котором ставится и решается вариационная задача. Здесь преодолеваются такие, например, затруднения: функционал $F[x(\cdot)]$ (1) определен на функциях, имеющих первую производную, но не обязательно имеющих вторую. Однако вторая производная появляется при вычислении функциональной производной. Сюда же относятся вопросы существования решения вариационной задачи: может оказаться, что существует точная нижняя грань $\inf_{x(\cdot)} F[x(\cdot)]$ (в том или ином функциональном пространстве), существует и минимизирующая последовательность $x^{(n)}(\cdot)$, т. е.

$$F[x^{(n)}(\cdot)] \rightarrow \inf_{x(\cdot)} F[x(\cdot)] \quad \text{при } n \rightarrow \infty,$$

*) В (3) можно вместо $O(\|\delta x\|^2)$ писать $o(\|\delta x\|)$, однако мы не будем стремиться к подобной общности.

однако предела $x^{(n)}(\cdot)$ ($n \rightarrow \infty$) нет, или же он есть, но лежит в другом функциональном пространстве, на элементах которого функционал типа (1) не определен.

Дальнейшее развитие классического вариационного исчисления привело к рассмотрению задач с более сложными функционалами, например, включающими старшие производные:

$$F[x(\cdot)] \equiv \int_0^T \Phi[x(t), \dot{x}(t), \ddot{x}(t)] dt.$$

Рассматривалась задача типа (1) с вектор-функцией $x(\cdot)$, краевые значения которой могли быть полностью или частично заданы, вариационные задачи для функций нескольких независимых переменных — в этом случае функционал вычислялся через значения частных производных искомой функции. Наконец, изучались изопериметрические задачи, в которых, кроме минимизируемого функционала $F[x(\cdot)]$, определялись функционалы $F_1[x(\cdot)]$, \dots , $F_m[x(\cdot)]$ того же типа (1), и на искомую функцию накладывались условия вида

$$F_i[x(\cdot)] = C_i, \quad i = 1, 2, \dots, m. \quad (6)$$

Основные типы задач, подходы к их решению и результаты были получены давно; они связаны с именами таких классиков естествознания, как Эйлер, Якоби, Вейерштрасс. Однако бурное развитие техники *) после второй мировой войны, характеризующееся, кроме всего прочего, четкой тенденцией к созданию оптимальных по своим качествам конструкций, привело к постановке ряда частных задач, которые были вариационными по существу дела, однако либо не укладывались в привычные рамки вариационного исчисления, либо это удавалось сделать ценой нежелательных искажений задачи. Постепенно выработались некоторые типичные формы новых вариационных задач, получившие имена пионеров этой области; так появились задачи Больца, Майера и другие. Отдавая должное этим ученым, мы не будем в дальнейшем пользоваться соответствующей терминологией, так как она отражает лишь историю становления современного вариационного исчисления, но не существо дела. Эти различные по наименованиям задачи не нуждаются ни в специфических методах теоретического исследования, ни в особых подходах при разработке алгоритмов их приближенного решения. Все эти задачи естественно укладываются в сложившуюся в настоящее время форму задачи оптимального управления, теоретический анализ которой не проще и не сложнее анализа упомянутых ее частных видов. Это же отно-

*) Особую роль для вариационного исчисления сыграло развитие автоматического управления и ракетостроения.

сится и к методам приближенного решения: хотя в оригинальных работах многие численные алгоритмы предлагались для тех или иных частных форм общей задачи оптимального управления, их переформулировка для общей задачи часто требует лишь редакционных изменений.

Постановка общей задачи оптимального управления естественно начинается с введения понятия *управляемой системы*. В дальнейшем этим термином обозначается некоторый объект, состояние которого в каждый момент t описывается вектор-функцией

$$x(t) = \{x^1(t), x^2(t), \dots, x^n(t)\}, \quad 0 \leq t \leq T. \quad (7)$$

Компоненты $x(t)$ называются *фазовыми координатами* управляемой системы, а n -мерное евклидово пространство E_n точек x — ее *фазовым пространством*. Эволюция состояния управляемой системы во времени определяется системой обыкновенных дифференциальных уравнений

$$\frac{dx}{dt} = f(x, u), \quad (8)$$

где $f(x, u) = \{f^1, f^2, \dots, f^n\}$ — известная n -мерная вектор-функция. Мы будем всюду в дальнейшем считать ее достаточно гладкой; вопросы, связанные с ослаблением предположений о гладкости $f(x, u)$, нас интересовать не будут. Поэтому в дальнейшем мы будем дифференцировать $f(x, u)$ столько раз, сколько потребуется при проведении выкладок. Далее,

$$u(t) = \{u_1, u_2, \dots, u_r\} \quad (9)$$

— некоторая вектор-функция, называемая *управлением*.

В данной задаче такие выражения, как «управлять системой», «выбрать управление», «определить управление» и тому подобное означают одно и то же — задать на *интервале управления* $[0, T]$ некоторую функцию $u(t)$. Естественно возникает вопрос о функциональном классе, из которого разрешается выбирать $u(t)$. Удобным оказался класс *измеримых функций*. С точки зрения теорем существования решений вариационных задач класс измеримых функций очень удобен. Однако при выводе необходимых условий оптимальности обычно происходит сужение задачи: сама исследуемая функция $u(t)$ предполагается не произвольной измеримой функцией, а гораздо более простой, например, кусочно непрерывной, кусочно гладкой, но ее разрешается подвергать вариациям $\delta u(t)$, относительно которых уже никаких предположений не делается: $\delta u(t)$ может быть произвольной измеримой функцией. Таким образом, объектом теоретических исследований является сравнительно простая функция $u(t)$, которая должна быть оптимальной в гораздо более широком множестве произволь-

ных (измеримых) функций $u(\cdot)$. Такая постановка вопроса не вызывает возражений с прикладной точки зрения: опыт решения содержательных задач показал, что их решения в подавляющем большинстве — сравнительно просто устроенные функции, имеющие разве лишь разрывы первого рода, а между точками разрывов — достаточно гладкие. (Иногда появляются точки сгущения точек разрыва.) Есть класс задач (и он достаточно содержателен), которые не имеют решения в классе кусочно непрерывных функций. Это — задачи с так называемыми *скользящими режимами*. (У этих задач нет решений и в классе измеримых функций.) Для подобных задач были разработаны методы их эквивалентной переформулировки, в которой в качестве управлений фигурируют некоторые новые функции, предположение о кусочной непрерывности которых уже достаточно естественно и приемлемо. В постановку современных вариационных задач обычно входит условие, ограничивающее допустимые значения $u(t)$. Его символическая запись имеет вид

$$u(t) \in U \text{ при всех } t \in [0, T], \quad (10)$$

где U — ограниченная замкнутая область r -мерного пространства E_r . Фактически эта область U задается системой неравенств вида

$$U: \quad \varphi_j(u) \leq 0, \quad j = 1, 2, \dots, J. \quad (10^*)$$

Измеримую вектор-функцию $u(t)$, удовлетворяющую (10), мы будем называть U -допустимым управлением. Оптимальное использование возможностей данной управляемой системы в рамках рассматриваемой задачи ограничивается выбором управления $u(\cdot)$. Условие (10) в содержательных задачах носит либо физический характер, либо связано с ограничениями технического порядка. В дополнение к системе (8) задается полный набор краевых условий. Мы будем символически обозначать этот набор

$$\Gamma(x) = 0.$$

Полнота его понимается в следующем смысле: при любой заданной функции $u(t)$ краевая задача

$$\frac{dx}{dt} = f[x, u(t)], \quad \Gamma(x) = 0, \quad (11)$$

имеет решение, и оно единственно. Обычно эти краевые условия имеют форму данных Коши, т. е. фактически $\Gamma(x) = 0$ имеет вид $x(0) - X_0 = 0$, однако встречаются и другие краевые задачи.

2. Функционалы, определенные на траекториях управляемой системы. Качество того или иного управления системой оценивается набором числовых характеристик — функционалов

$F_i [u(\cdot)], i=0, 1, \dots, m$. В приложениях они часто называются показателями качества, критериями качества и т. д. Рассмотрим некоторые типичные формулы фактического вычисления функционалов:

$$F [u(\cdot)] \equiv \int_0^T \Phi [x(t), u(t)] dt, \quad (12)$$

$$F [u(\cdot)] \equiv \Phi [x(t')], \quad t' \text{ — заданная точка на } [0, T], \quad (13)$$

$$F [u(\cdot)] \equiv \max_{t \in [0, T]} \Phi [x(t)], \quad (14)$$

$$F [u(\cdot)] \equiv \int_0^T |\Phi [x(t), u(t)]| dt, \quad (15)$$

$$F [u(\cdot)] \equiv \operatorname{vrai} \max_t \Phi [x(t), u(t)]^* \quad (16)$$

Φ в правых частях этих определений считается заданной достаточно гладкой функцией своих аргументов. Не претендуя на исчерпывающую полноту, ограничимся пока этими конструкциями. Их, а также гладких функций от функционалов перечисленных типов, достаточно для постановки большинства прикладных задач. В дальнейшем будут использоваться и другие конструкции функционалов. В формулах (12)–(16) фазовая траектория $x(\cdot)$ связана с управлением краевой задачей (11) и однозначно определяется им. Этим оправдывается обозначение выражений в правых частях определений через $F [u(\cdot)]$. Фактическое вычисление $F [u(\cdot)]$ требует решения краевой задачи (11), для чего используются соответствующие приближенные методы, ориентированные, как правило, на использование ЭВМ. Выбор того или иного численного алгоритма определяется содержательным характером краевой задачи (11). Особых трудностей при этом не возникает, так как задача оптимизации какого-либо объекта обычно ставится после того, как расчет его функционирования при каком-то фиксированном управлении уже достаточно освоен, и подходящие численные методы разработаны и проверены.

Заметим, что управление $u(\cdot)$ определено, по существу, с точностью до значений на множестве меры нуль; управления $u(\cdot)$ и $\tilde{u}(\cdot)$, отличающиеся лишь на множестве нулевой меры, определяют одно и то же движение управляемой системы, одну и ту же траекторию $x(t)$. Поэтому в постановке задачи не может

*) Приставка *vrai* означает, что имеется в виду существенный максимум $\Phi [x(t), u(t)]$, значение которого не изменится, если t пробегает не весь отрезок $[0, T]$, а любое множество M , получающееся из $[0, T]$ удалением произвольного множества меры нуль.

фигурировать функционал типа $F[u(\cdot)] \equiv \Phi[u(t')]$, так как значение $u(\cdot)$ в одной точке t' может быть взято произвольным, и это никак не повлияет на поведение системы. В литературе, тем не менее, иногда встречаются задачи с такими функционалами, что свидетельствует о недостаточно продуманной постановке задач, о том, что, возможно, забыто условие принадлежности $u(t)$ к некоторому классу непрерывных функций *). Например, добавив к условиям задачи требование выполнения условия Липшица $\|u(t_2) - u(t_1)\| \leq C |t_2 - t_1|$, превратим конструкцию $F[u(\cdot)] \equiv \Phi[u(t')]$ в содержательно осмысленную и имеющую право участвовать в постановке вариационной задачи.

Теперь в нашем распоряжении есть все для того, чтобы могла быть сформулирована типичная задача оптимального управления.

Для управляемой системы:

$$\frac{dx}{dt} = f(x, u), \quad \Gamma(x) = 0, \quad 0 \leq t \leq T, \quad (I)$$

нужно определить управление $u(\cdot)$, минимизирующее значение функционала F_0 :

$$\min_{u(\cdot)} F_0[u(\cdot)]. \quad (II)$$

При этом должно быть выполнено условие

$$u(t) \in U \text{ при всех } t \in [0, T] \quad (III)$$

и дополнительные ограничения

$$F_i[u(\cdot)] = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m. \quad (IV)$$

Здесь $F_i[u(\cdot)]$, $i = 0, 1, \dots, m$, — функционалы, каждый из которых может иметь вид (12) — (16), или быть функцией таких функционалов.

Сформулированная выше задача не является самой общей; ряд простых обобщений ее будет обсужден ниже (см. § 7); однако и в такой форме она включает в себя ряд задач, обычно трактуемых отдельно друг от друга. Рассмотрим их.

Задача с ограничением в фазовом пространстве. Пусть на управление $u(\cdot)$ наложено условие: порождаемая им фазовая траектория $x(t)$ обязана находиться в некоторой заданной области R фазового пространства:

$$x(t) \in R \text{ при всех } t \in [0, T]. \quad (17)$$

*) Если подобная задача все же решается каким-либо численным методом, то «забытое» условие явно или неявно вводится в алгоритм (см. [44]).

Фактически (17) реализуется заданием одного или нескольких неравенств типа

$$G[x(t)] \leq 0 \text{ при всех } t, \quad (18)$$

с известной гладкой функцией $G(\xi)$. Определив функционал

$$F[u(\cdot)] \equiv \max_t G[x(t)],$$

включим в IV дополнительное условие $F[u(\cdot)] \leq 0$.

Задачи с ограничениями общего типа. Эти задачи, для которых иногда используется исторически сложившееся название *задачи на узкие места*, ставятся обычно в двух внешне различных формах. Именно, в условия вводится требование

$$u(t) \in U[x(t)] \text{ при всех } t,$$

т. е. вид области U зависит от $x(t)$, обычно непрерывно*). В другой форме задача дополняется условием

$$\{u(t), x(t)\} \in Q \text{ при всех } t, \quad (18^*)$$

где Q — некоторая ограниченная замкнутая область в пространстве $E_r \times E_n$. Фактическая реализация в обоих случаях означает задание одного или нескольких неравенств вида

$$\Phi[x(t), u(t)] \leq 0 \text{ при всех } t, \quad (19)$$

которые очевидным образом вводятся в дополнительные условия IV с функционалами типа (16).

Заметим, что сведение задачи оптимального управления к классической вариационной задаче путем разрешения соотношений $\dot{x} = f(x, u)$ относительно u и исключения u из правых частей формул для функционалов типа (12)—(16), как правило, неосуществимо, и не столько в силу технических трудностей такого исключения, сколько из-за более веских причин.

Во-первых, неясно, как при таком исключении сохранить условие $u \in U$, не имеющее аналога в классической задаче; во-вторых, как правило, размерность вектора u меньше размерности вектора x . В задачу оптимального управления $u(\cdot)$ и $x(\cdot)$ входят существенно неравноправно: если более или менее любой функции $u(\cdot)$ соответствует некоторая траектория $x(t)$, то почти любой дифференцируемой функции $x(t)$ никакого $u(t)$, удовлетворяющего уравнению $\dot{x} = f(x, u)$, не соответствует. Этот факт определяет выбор $u(\cdot)$ в качестве независимого аргумента задачи и сказывается самым серьезным образом при конструировании вычис-

*) Непрерывность понимается в следующем смысле: $U(x + \delta x) \subset U_{\delta} \subset U_{\|\delta x\|}(x)$, где U_{δ} — расширение области U сферами радиуса δ с центрами во всех точках U .

лительных алгоритмов. В то же время классическая вариационная задача очевидным образом формулируется как задача оптимального управления — определить $u(\cdot)$ из условий

$$\min_{u(\cdot)} F_0[u(\cdot)],$$

где $F_0[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt$, на решениях системы $\dot{x} = u$, $x(0) = X_0$ при дополнительном условии $x(T) = X_1$.

Другим важным обстоятельством, определяющим неклассический характер задачи оптимального управления, является наличие в задаче условий типа неравенств. Это — условия $u(t) \in U$, условия (17), (18). Они, как показал опыт решения таких задач, весьма существенны: снятие подобных условий обычно полностью лишает задачу содержательной ценности, так как приводит к решениям либо физически нелепым, либо неприемлемым по техническим условиям. Как правило, в оптимальном решении имеются как интервалы времени, на которых реализуется знак равенства, так и интервалы, на которых реализуется строгое неравенство; на первых условие может быть заменено привычным для классического вариационного исчисления условием типа равенства, на последних — снято. К сожалению, расположение и размеры этих интервалов выясняются лишь после решения задачи. Это обстоятельство также имеет глубокие последствия в вопросах конструирования численных методов: классический вычислительный аппарат линейной алгебры становится неэффективным и заменяется более соответствующим характеру современных вариационных задач вычислительным аппаратом линейного (и нелинейного) программирования.

§ 3. Дифференцирование функционалов, определенных на траекториях управляемой системы

Основным инструментом теоретического анализа задач оптимального управления и конструирования методов их приближенного решения является вычисление функциональных производных от входящих в постановку задачи функционалов. В настоящее время сложилась сравнительно стандартная техника дифференцирования функционалов, определенных на траекториях управляемой системы. Изложим ее, ограничившись первыми двумя типичными конструкциями

$$F[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt, \quad (1)$$

$$F[u(\cdot)] \equiv \Phi[x(t')]. \quad (2)$$

Что касается остальных трех — (14), (15), (16), то они, вообще говоря, не имеют производных Фреше. Они дифференцируемы в некотором специальном смысле — по направлениям в функциональном пространстве. В § 4 будет дано соответствующее определение и получены производные по направлениям для этих конструкций функционалов. Вернемся к функционалам (1) и (2), заметив, что нижеследующие вычисления легко превращаются в доказательство дифференцируемости их в смысле Фреше.

Пусть управление $u(\cdot)$ возмущено малой функцией $\delta u(\cdot)$; $\|\delta u(\cdot)\|$ считается малой величиной первого порядка, где, по определению,

$$\|\delta u(\cdot)\| \equiv \max_{i=1, 2, \dots, r} \max_{0 \leq t \leq T} |\delta u_i(t)|.$$

Следствием этого является соответствующее малое возмущение фазовой траектории: $x(t)$ переходит в $x(t) + \delta x(t)$, причем $\|\delta x(\cdot)\| \equiv O(\|\delta u(\cdot)\|)$ и $\delta x(t)$ является решением уравнения в вариациях:

$$\frac{d\delta x}{dt} = f_x[t] \delta x + f_u[t] \delta u; \quad \Gamma_x \delta x = 0. \quad (3)$$

Это уравнение определено на невозмущенной траектории $\{u(\cdot), x(\cdot)\}$; здесь и в дальнейшем приняты обозначения типа $f_x[t] \equiv f_x[x(t), u(t)]$, $f_u[t] \equiv f_u[x(t), u(t)]$, $\Gamma_x \delta x = 0$ — символическая запись краевых условий для δx ; они получаются простым варьированием краевых условий $\Gamma(x) = 0$ *). Заметим, что можно было бы использовать уравнение в вариациях в форме

$$\frac{d\delta x}{dt} = f_x[t] \delta x + f_u[t] \delta u + O(\|\delta u\|^2), \quad \Gamma_x \delta x + O(\|\delta u\|^2) = 0, \quad (3^*)$$

и считать $\delta x(t)$ точной разностью между возмущенной и невозмущенной траекториями; однако проще будет пользоваться формой (3), не забывая, что в этом случае $\delta x(t)$ отличается от точной разности на $O(\|\delta u\|^2)$. Разумеется, мы считаем, что все условия, обеспечивающие обоснованность используемой здесь теории возмущений, выполнены. Таким образом, $\delta x(t)$ однозначно определяется возмущением управления $\delta u(\cdot)$ как решение краевой задачи (3). Вычислим теперь приращения функционалов. Прямое варьирование формул (1) и (2) дает

$$\begin{aligned} \text{для (1):} \quad F[u(\cdot) + \delta u(\cdot)] &= F[u(\cdot)] + \int_0^T \Phi_x[t] \delta x(t) dt + \\ &+ \int_0^T \Phi_u[t] \delta u(t) dt + O(\|\delta u\|^2), \end{aligned} \quad (4)$$

*) Если, например, условие $\Gamma(x) = 0$ имеет вид $x(0) - X_0 = 0$ (данные Коши), то $\Gamma_x \delta x = 0$ есть условие вида $\delta x(0) = 0$.

для (2): $F[u(\cdot) + \delta u(\cdot)] = F[u(\cdot)] + \Phi_x[x(t')] \delta x(t') +$
 $+ O(\|\delta u\|^2) = F[u(\cdot)] + \int_0^T \Phi_x[t'] \delta x(t) \delta(t - t') dt + O(\|\delta u\|^2). \quad (5)$

Здесь $\delta(t - t')$ — δ -функция Дирака с полюсом в t' . Обе формулы запишем в общем виде:

$$\delta F[\delta u(\cdot)] = \int_0^T \tilde{w}(t) \delta u(t) dt + \int_0^T Y(t) \delta x(t) dt. \quad (6)$$

Здесь $\tilde{w}(t)$ и $Y(t)$ — вектор-функции размерности r и n соответственно, определенные, вообще говоря, на невозмущенной траектории. Произведение $\tilde{w} \delta u$ есть, разумеется, скалярное произведение; этой формой его записи мы будем пользоваться наряду с общепринятой $(w, \delta u)$. Если для главной части приращения функционала может быть получена формула (6), он оказывается дифференцируемым. Правда, следует оговорить свойства функций $\tilde{w}(t)$, $Y(t)$.

1. В функции $Y(t)$ допустимы особенности типа δ -функции. Это связано с тем, что $\delta x(t)$ есть решение уравнения в вариациях и является непрерывной функцией. Поэтому интеграл типа $\int \delta(t - t') \delta x(t) dt$ имеет смысл.

2. $\delta u(t)$ есть произвольная ограниченная функция, поэтому в $\tilde{w}(t)$ особенности типа δ -функции недопустимы. Не стремясь к максимальной общности, будем считать $\tilde{w}(t)$ произвольной ограниченной функцией.

3. Из уравнения (3) видно, что $\delta \dot{x}(t)$ есть функция того же типа гладкости, что и $\delta u(t)$; поэтому в общем случае в $Y(t)$ недопустимы особенности типа $\delta_t(t - t')$.

Однако иногда это оказывается возможным; пусть, как это часто бывает в прикладных задачах, уравнение для одной из компонент x не содержит управления: $\dot{x}^1 = f^1(x)$. Тогда в постановке вариационной задачи допустим функционал $F[u(\cdot)] \equiv \Phi[x^1(t')]$, варьирование которого приводит к формуле (6) с $Y(t)$, имеющей особенность типа производной δ -функции. В этом случае $\delta \dot{x}^1 = f_x^1 \delta x$ оказывается непрерывной функцией t и соответствующий интеграл в (6) имеет смысл. Мы все же не будем рассматривать подобных обобщений, так как в этом случае функционалу легко придается стандартная форма (2):

$$\Phi[x^1(t')] = \Phi[f^1(x(t'))].$$

Для вычисления производной функционала нужно, используя уравнение в вариациях, исключить $\delta x(\cdot)$ из (6) и перейти к формуле типа

$$\delta F[\delta u(\cdot)] = \int_0^{T_1} w(t) \delta u(t) dt.$$

Проще всего это достигается использованием известного *тождества Лагранжа*.

Пусть $\delta x(t)$ непрерывная вектор-функция с кусочно непрерывной производной, а $\psi(t)$ — вектор-функция, которая может иметь конечное число разрывов первого рода (ниже для простоты предположим, что $\psi(t)$ имеет лишь один разрыв в точке t'). Тогда имеет место тождество *)

$$\int_0^T \left\{ \psi \left(\frac{d\delta x}{dt} - f_x[t] \delta x \right) + \delta x \left(\frac{d\psi}{dt} + f_x^*[t] \psi \right) \right\} dt = \psi \delta x \Big|_0^T. \quad (7)$$

Доказательство состоит из двух утверждений:

1) $(\psi, f_x \delta x) \equiv (f_x^* \psi, \delta x)$,

2) там, где $\psi(t)$ и $\delta x(t)$ дифференцируемы, $\psi \frac{d\delta x}{dt} + \delta x \frac{d\psi}{dt} = \frac{d}{dt}(\psi, \delta x)$.

Разрыв в $\psi(t)$ порождает δ -функцию в ψ с полюсом в t' и с интенсивностью $\psi(t' + 0) - \psi(t' - 0)$. В этом случае левая часть (7) есть

$$\begin{aligned} \lim_{\varepsilon \rightarrow 0} \left\{ \int_0^{t'-\varepsilon} \frac{d}{dt}(\psi, \delta x) dt + \int_{t'+\varepsilon}^T \frac{d}{dt}(\psi, \delta x) dt + \int_{t'-\varepsilon}^{t'+\varepsilon} \psi \delta \dot{x} dt + \int_{t'-\varepsilon}^{t'+\varepsilon} \psi \dot{\delta x} dt \right\} = \\ = \psi(t' - 0) \delta x(t') - \psi(0) \delta x(0) + \psi(T) \delta x(T) - \\ - \psi(t' + 0) \delta x(t') + \delta x(t') [\psi(t' + 0) - \psi(t' - 0)] = \psi \delta x \Big|_0^T. \end{aligned}$$

Теперь, заменив $\delta \dot{x} - f_x \delta x$ на $f_x^*[t] \delta u(t)$ (из уравнения в вариациях) и конкретизировав функцию $\psi(t)$ как решение краевой задачи

$$\frac{d\psi}{dt} + f_x^*[t] \psi = -Y(t); \quad \Gamma_x^* \psi = 0, \quad (8)$$

сопряженной к краевой задаче для $\delta x(t)$, получим

$$\int_0^T Y(t) \delta x(t) dt = \int_0^T (\psi(t), f_x^*[t] \delta u(t)) dt,$$

и окончательно имеем формулу для вариации функционала

$$\delta F[\delta u(\cdot)] = \int_0^T \tilde{w}(t) \delta u(t) dt + \int_0^T (f_x^*[t] \psi(t), \delta u(t)) dt. \quad (9)$$

*) Звездочкой будем отмечать сопряженную матрицу, сопряженный оператор.

Таким образом,

$$\frac{\partial F[u(\cdot)]}{\partial u(t)} = \tilde{w}(t) + f_u^*[t]\psi(t). \quad (10)$$

В (8) $\Gamma_x^*\psi = 0$ есть символическая запись краевых условий, сопряженных к условиям $\Gamma_x\delta x = 0$; конкретная их форма легко определяется требованием: из $\Gamma_x\delta x = 0$ и $\Gamma_x^*\psi = 0$ должно следовать $(\psi, \delta x)|_0^T = 0$.

Примеры сопряженных условий:

1. Пусть $\Gamma(x) = 0$ — данные Коши: $x(0) - X_0 = 0$; тогда $\Gamma_x\delta x = 0$ — тоже данные Коши: $\delta x(0) = 0$, а $\Gamma_x^*\psi = 0$ — данные Коши на правом конце: $\psi(T) = 0$.

2. Сопряженными к условиям периодичности $\delta x(0) - \delta x(T) = 0$, очевидно, являются также условия периодичности $\psi(0) - \psi(T) = 0$.

3. Сопряженные к общим краевым условиям вида $\Gamma_x\delta x = 0$: $A\delta x(0) + B\delta x(T) = 0$, где A, B — заданные матрицы *) $n \rightarrow n$, не обязательно имеющие обратные, можно получить так: образуем $2n$ -мерные векторы $z = \{\delta x(0), \delta x(T)\}$, $\xi = \{\psi(0), -\psi(T)\}$ и матрицу $2n \rightarrow n$: $C = \{A, B\}$. Краевые условия, которые теперь могут быть записаны в виде $Cz = 0$, предположим невырожденными. Это означает, что можно выбрать n столбцов, образующих невырожденную матрицу C_1 ; выделив соответствующие компоненты z в n -вектор z_1 , а остальные — в вектор z_2 , запишем краевые условия в виде $Cz \equiv C_1z_1 + C_2z_2 = 0$. Таким образом, общий вид векторов z , удовлетворяющих условию $Cz = 0$, есть $\{-C_1^{-1}C_2z_2, z_2\}$, где z_2 — произвольный n -вектор. Тогда из условия $\psi\delta x|_0^T = 0$ следует, что $(z, \xi) = (z_1, \xi_1) + (z_2, \xi_2) = -(C_1^{-1}C_2z_2, \xi_1) + (z_2, \xi_2) = -(z_2, C_2^*C_1^{-1}\xi_1) + (z_2, \xi_2) = (z_2, \xi_2 - C_2^*C_1^{-1}\xi_1) = 0$. Поскольку z_2 произволен, то сопряженные условия имеют вид

$$\xi_2 = C_2^*C_1^{-1}\xi_1.$$

4. Заслуживает пояснения следующий случай, встречающийся иногда в приложениях: пусть для решения уравнения $\dot{x} = f(x)$ поставлено условие $x(T/2) = 0$; тогда для сопряженной функции $\psi(t)$ получим уравнение $\dot{\psi} + f_x^*[t]\psi = -Y$ с краевыми условиями в избыточном числе: $\psi(0) = 0$; $\psi(T) = 0$. Однако избыточность условий не приводит к противоречию, так как в $\psi(t)$ допустим произвольный разрыв в точке $t = T/2$.

В приложениях часто встречается функционал $F[u(\cdot)] \equiv \Phi[x(T)]$ ($\Gamma(x) = 0$ в этом случае обычно суть данные Коши в точке $t = 0$: $x(0) = X_0$). Используя описанную выше схему вычисления его производной, получим в правой части уравнения (8) δ -функцию на конце интервала; это, как известно, приводит к неоднородности краевых условий. Однако проще прямо получить необходимый результат. Исходя из равенства

$$\delta F[\delta u(\cdot)] = \Phi_x[x(T)]\delta x(T)$$

*) Для размера матриц, отображающих n -мерное пространство в r -мерное, будем использовать обозначение $n \rightarrow r$.

и положив в (8) $Y(t) = 0$, из тождества Лагранжа получаем

$$\psi(T) \delta x(T) = \int_0^T \psi(t) f_u[t] \delta u(t) dt.$$

Достаточно в этом случае в качестве краевых условий для ψ взять данные Коши при $t = T$: $\psi(T) = \Phi_x[x(T)]$, чтобы получить вариацию

$$\delta F[\delta u(\cdot)] = \int_0^T \psi(t) f_u[t] \delta u(t) dt,$$

т. е.

$$\frac{\partial F[u(\cdot)]}{\partial u(\cdot)} = f_u^*[t] \psi(t). \quad (11)$$

§ 4. Функционалы, дифференцируемые по направлениям в функциональном пространстве

Здесь будут изучены функционалы

$$F[u(\cdot)] \equiv \max_t \Phi[x(t)], \quad (1)$$

$$F[u(\cdot)] \equiv \int_0^T |\Phi[x(t)]| dt. \quad (2)$$

По некоторым причинам, которые будут разъяснены в дальнейшем, мы пока не рассматриваем функционалы типа $\max_t \Phi[x(t), u(t)]$.

1. Рассмотрим сначала конструкцию (1). Пусть управление $u(\cdot)$ получило малое возмущение и перешло в $u(\cdot) + \delta u(\cdot)$; следствием этого является малое возмущение фазовой траектории: $x(t)$ перешло в $x(t) + \delta x(t)$ и

$$\begin{aligned} F[u(\cdot) + \delta u(\cdot)] &= \max_t \Phi[x(t) + \delta x(t)] = \\ &= \max_t \{\Phi[x(t)] + \Phi_x[x(t)] \delta x(t) + O(\|\delta u\|^2)\}. \end{aligned} \quad (3)$$

Выделим на $[0, T]$ множество M условием*)

$$t \in M, \text{ если } \Phi[x(t)] = F[u(\cdot)]. \quad (4)$$

Ясно, что $t \notin M$ можно не рассматривать: если $\Phi[x(t)] < F[u(\cdot)]$, то при достаточно малом значении $\|\delta u\|$ максимум в (3) в этой точке

*) Множество M замкнуто в силу непрерывности $\Phi[x(t)]$.

достигаться не может. Поэтому от (3) мы перейдем к выражению для первой вариации (пренебрегая $O(\|\delta u\|^2)$):

$$\begin{aligned} F[u(\cdot)] + \delta F[\delta u(\cdot)] &= \max_{t \in M} \{ \Phi[x(t)] + \Phi_x[t] \delta x(t) \} = \\ &= F[u(\cdot)] + \max_{t \in M} \Phi_x[t] \delta x(t). \end{aligned}$$

Таким образом,

$$\delta F[\delta u(\cdot)] = \max_{t \in M} \Phi_x[t] \delta x(t). \quad (5)$$

Связь между возмущением управления $\delta u(\cdot)$ и порожденным им возмущением фазы $\delta x(t)$ носит (с точностью до $O(\|\delta u\|^2)$) линейный характер: если возмущения $\delta u_1(\cdot)$ и $\delta u_2(\cdot)$ порождают соответственно $\delta x_1(t)$ и $\delta x_2(t)$ (ниже мы выпишем формулы вычисления $\delta x(t)$ через $\delta u(\cdot)$), то $\delta u(\cdot) = \alpha \delta u_1(\cdot) + \beta \delta u_2(\cdot)$ порождает $\delta x(t) = \alpha \delta x_1(t) + \beta \delta x_2(t)$. Однако правая часть (5) не является линейным функционалом от $\delta x(\cdot)$, поскольку

$$\max_{t \in M} \Phi_x(t) [\delta x_1(t) + \delta x_2(t)] \leq \max_{t \in M} \Phi_x \delta x_1 + \max_{t \in M} \Phi_x \delta x_2,$$

и знак точного равенства гарантируется только в том случае, когда $\delta x_2(t) = \lambda \delta x_1(t)$, $\lambda \geq 0$.

Введем определение: Функционал $F[u(\cdot)]$ называется *дифференцируемым по направлениям* в функциональном пространстве (иначе, — дифференцируемым в смысле Гато), если для любой функции $v(\cdot)$ (из того же пространства, что и $u(\cdot)$) имеет место формула

$$F[u(\cdot) + sv(\cdot)] = F[u(\cdot)] + sD[u(\cdot), v(\cdot)] + o(s), \quad (6)$$

где s — любое малое неотрицательное число, а D — некоторый функционал от $u(\cdot)$ и $v(\cdot)$, называемый *производной функционала F в точке $u(\cdot)$ по направлению $v(\cdot)$* .

Из (5) имеем

$$\delta F[sv(\cdot)] = \max_{t \in M} \Phi_x[t] sy(t) = s \max_{t \in M} \Phi_x[t] y(t),$$

где $sy(t)$ есть возмущение фазы $x(t)$ за счет возмущения управления $\delta u(\cdot) = sv(\cdot)$. Сопоставляя это выражение с (6), получаем предварительную формулу для производной Гато функционала (1):

$$D[u(\cdot), v(\cdot)] = \max_{t \in M} \Phi_x[x(t)] y(t). \quad (7)$$

Окончательная формула получится после исключения $y(t)$ с помощью $v(\cdot)$. Этим мы сейчас и займемся, воспользовавшись известным уже результатом: для выражения в фиксированной точке

$t' \in M \subset [0, T]$ величины $\Phi_x[t'] \delta x(t')$ через порождающее $\delta x(t)$ возмущение управления $\delta u(\cdot)$ следует решить краевую задачу

$$\frac{d\psi(t, t')}{dt} + f_x^*[t] \psi(t, t') = -\Phi_x[t'] \delta(t - t'),$$

$$\Gamma_x^* \psi = 0, \quad (8)$$

после чего

$$\Phi_x[x(t')] \delta x(t') = \int_0^T \psi(t, t') f_u[t] \delta u(t) dt. \quad (9)$$

Имея функцию $\psi(t, t')$, определенную на $[0, T] \times M$, преобразуем (7) к окончательной форме

$$D[u(\cdot), v(\cdot)] = \max_{t' \in M} \int_0^T \psi(t, t') f_u[t] v(t) dt. \quad (10)$$

Фактическое вычисление (численное, например) производной Гато (10) существенно сложнее вычисления производных Фреше для функционалов, рассмотренных в § 3: вычисление и использование последних требует однократного решения краевой задачи типа (3.8) и запоминания функции одного переменного $\psi(t)$. Для того чтобы работать с производной Гато, нужно вычислить и запомнить функцию двух переменных $\psi(t, t')$. Вводя на M некоторую достаточно плотную конечную сетку t'_1, t'_2, \dots, t'_i , мы можем получить достаточно точную аппроксимацию производной Гато после l -кратного решения краевых задач типа (8), запомнив функции $\psi(t, t'_1), \psi(t, t'_2), \dots, \psi(t, t'_i)$. Хотя эта процедура отпугивает своей громоздкостью, именно она использовалась автором в многочисленных расчетах; в сочетании с некоторыми дополнительными приемами, этот подход позволил эффективно решить ряд сложных задач с функционалами типа (1), причем расход машинного времени был сравнительно невелик. Теперь обсудим одну нестрогость, допущенную в проведенном выше анализе. Речь идет о переходе

$$\max_{0 \leq t \leq T} \{\Phi[x(t)] + \Phi_x[t] \delta x(t)\} = \max_{t \in M} \{\Phi[x(t)] + \Phi_x[t] \delta x(t)\}. \quad (11)$$

Строго говоря, он неверен для сколь угодно малого, но все же конечного $\delta x(t)$. Исправить формулу (11) можно, используя в ее правой части не M , а множество $M(\|\delta u\|)$, определенное свойством:

$$t \in M(\|\delta u\|), \quad \text{если} \quad \Phi[x(t)] \geq F[u(\cdot)] - 2C\|\delta u\|, \quad (12)$$

где постоянная C выбрана так, чтобы обеспечивалось неравенство $|\Phi_x[t] \delta x(t)| < C\|\delta u\|$, что, разумеется, возможно ввиду линейной зависимости $\delta x(t)$ от $\delta u(\cdot)$. Пусть теперь $\delta u(\cdot) = sv(\cdot)$, $\delta x(t) =$

$= sy(t)$, причем $y(t)$ связан с $v(\cdot)$ уравнением в вариациях, а $s > 0$ — малый параметр; обозначим $M_s = M(s \| v \|)$ и докажем, что

$$\max_{t \in M_s} \{ \Phi[x(t)] + s\Phi_x[t] y(t) \} = \max_{t \in M} \{ \Phi[t] + s\Phi_x[t] y(t) \} + o(s). \quad (13)$$

Этим упомянутая нестрогость будет устранена. Для доказательства используем следующие простые факты.

1) Обозначим через $t_s \in M_s$ какую-нибудь точку, в которой достигается максимум в левой части (13). Тогда при достаточно малых s найдется точка $t_s^* \in M$ такая, что $|t_s - t_s^*| = \eta(s)$, $\lim_{s \rightarrow 0} \eta(s) = 0$ (t_s^* — ближайшая к t_s точка в M). Это легко устанавливается рассуждением от противного: предположив существование $s_1 > s_2 > \dots \rightarrow 0$, для которых $\eta(s_i) \geq a > 0$, и взяв t^* — какую-нибудь предельную точку последовательности t_{s_1}, t_{s_2}, \dots , получим противоречие, так как $\Phi[x(t^*)] = F[u(\cdot)]$, и те из t_{s_i} , пределом которых является $t^* \in M$, не могут отстоять от M на расстоянии, большем $a > 0$. А то, что $t^* \in M$, легко усматривается из определения M_s (12):

$$\text{из } t_s \in M_s \text{ следует } \Phi[x(t_s)] > F[u(\cdot)] - O(s).$$

2) Функция $y(t)$ непрерывна, ограничена и имеет ограниченную производную, так как является решением линейного уравнения в вариациях. Пусть t_s^* — ближайшая к t_s точка M , $\Delta_s = t_s - t_s^*$, $|\Delta_s| \leq \eta(s)$. Тогда

$$\begin{aligned} \max_{t \in M_s} \{ \Phi[t] + s\Phi_x[t] y(t) \} &= \Phi[t_s] + s\Phi_x[t_s] y(t_s) = \\ &= \Phi[t_s^* + \Delta_s] + s\Phi_x[t_s^* + \Delta_s] y(t_s^* + \Delta_s) \leq \\ &\leq \Phi[t_s^*] + s\Phi_x[t_s^*] y(t_s^*) + sO(\Delta_s) \leq \\ &\leq \max_{t \in M} \{ \Phi[t] + s\Phi_x[t] y(t) \} + sO(\Delta_s) \end{aligned}$$

(было использовано соотношение $\Phi[t_s^* + \Delta_s] = \Phi[t_s] \leq \Phi[t_s^*]$, так как $t_s^* \in M$). Учитывая, что $M \subseteq M_s$, получим

$$\max_{t \in M} \{ \Phi + s\Phi_w y \} \leq \max_{t \in M_s} \{ \Phi + s\Phi_x y \} \leq \max_{t \in M} \{ \Phi + s\Phi_x y \} + s\eta(s).$$

Таким образом, доказано соотношение (13) и вычисление производной Гато функционала получило достаточно строгое обоснование. Что касается $\eta(s)$, то получить более сильную оценку, чем $\eta(s) \rightarrow 0$ при $s \rightarrow 0$, нельзя в общем случае. Это легко понять по рис. 1: если функция $\Phi[x(t)]$ выходит на максимум с касанием k -го порядка ($k > 1$), а $\Phi_x[t] y(t)$ имеет вид, качественно изображенный на том же

рис. 1, то t_s^* есть крайняя левая (на рисунке) точка M_s и $|t_s - t_s^*| = O(s^{1/k})$.

Несложный анализ изображенной на рисунке ситуации показывает неулучшаемость оценки (14) по порядку величин.

Заметим, наконец, что в случае, когда M состоит только из одной точки t' , функционал (1) оказывается дифференцируемым по Фреше *).

Весь проведенный выше анализ существенно использовал свойства гладкости функций $x(t)$ и $y(t)$, которыми они обладают

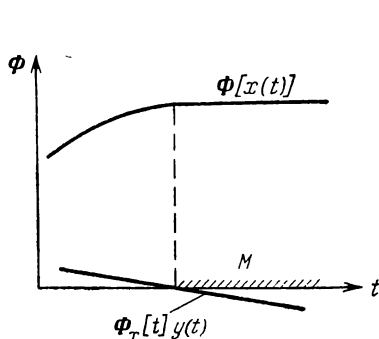


Рис. 1.

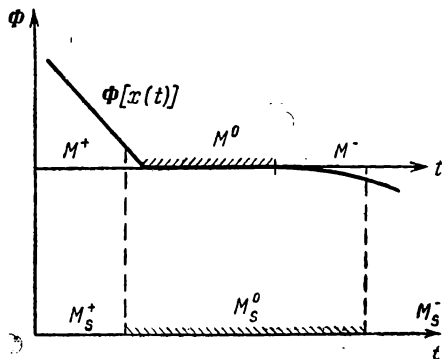


Рис. 2.

как решения дифференциальных уравнений с ограниченными правыми частями.

Функции $u(t)$, $v(t)$ подобных свойств не имеют, поэтому здесь не рассматриваются функционалы вида

$$F[u(\cdot)] = \max_t \Phi[x(t), u(t)].$$

Их анализ требует привлечения более тонких средств.

2. Вычисление производной Гато функционала

$$F[u(\cdot)] \equiv \int_0^T |\Phi[x(t)]| dt$$

осуществляется аналогичным образом. Пусть функция $\Phi[x(t)]$ имеет вид, качественно изображенный на рис. 2. Разобьем $[0, T]$ на три множества M^0 , M^- , M^+ :

$$\begin{aligned} t \in M^0, & \quad \text{если } \Phi[x(t)] = 0, \\ t \in M^-(M^+), & \quad \text{если } \Phi[x(t)] < 0 \quad (> 0). \end{aligned}$$

*) Разумеется, при условии, что точка максимума t' , как функционал от $u(\cdot)$, дифференцируема.

Пусть управление возмущено функцией $sv(\cdot)$, соответствующее возмущение фазовой траектории системы есть $sy(t)$. Тогда

$$\begin{aligned} F[u(\cdot) + sv(\cdot)] &= \int_0^T |\Phi[x(t)] + s\Phi_x[x(t)]y(t)| dt + O(s^2) = \\ &= s \int_{M^0} |\Phi_x[x(t)]y(t)| dt + \int_{M^+} \{\Phi[x(t)] + s\Phi_x[t]y(t)\} dt - \\ &\quad - \int_{M^-} \{\Phi[x(t)] + s\Phi_x[t]y(t)\} dt + O(s^2). \end{aligned} \quad (14)$$

В этом выводе допущена некоторая нестрогость; она аналогична рассмотренной при анализе функционала (1), и мы разберемся с ней, сформулировав сначала результат, очевидным образом следующий из (14): производная Гато функционала (2) вычисляется по формуле

$$\begin{aligned} D[u(\cdot), v(\cdot)] &= \int_{M^0} |\Phi_x[t]y(t)| dt + \int_{M^+} \Phi_x[t]y(t) dt - \\ &\quad - \int_{M^-} \Phi_x[t]y(t) dt. \end{aligned} \quad (15)$$

Она пока имеет предварительный характер, так как предстоит еще заменить $y(t)$ его выражением через возмущение независимого аргумента $v(\cdot)$. Аппарат исключения уже подготовлен, и мы сразу перейдем к результату. Определим функции $\psi(t)$ и $\psi(t, t')$ решением краевых задач

$$\frac{d\psi(t)}{dt} + f_x^*[t]\psi(t) = -Y(t); \quad \Gamma_x^*\psi = 0, \quad (16)$$

где

$$\begin{aligned} Y(t) &= \begin{cases} \Phi_x[t], & t \in M^+, \\ 0, & t \in M^0, \\ -\Phi_x[t], & t \in M^-, \end{cases} \\ \frac{d\psi(t, t')}{dt} + f_x^*[t]\psi(t, t') &= -\Phi_{xx}^*[t']\delta(t - t'), \\ \Gamma_x^*\psi &= 0; \quad t' \in M^0. \end{aligned}$$

Тогда

$$\begin{aligned} \int_{M^+} \Phi_x y dt - \int_{M^-} \Phi_x y dt &= \int_0^T \psi(t) f_u[t] v(t) dt, \\ \int_{M^0} |\Phi_x[t']y(t')| dt' &= \int_{M^0} \left| \int_0^T \psi(t, t') f_u[t] v(t) dt \right| dt', \end{aligned}$$

и окончательная формула для производной по направлению $v(\cdot)$ в точке $u(\cdot)$ имеет вид

$$D[u(\cdot), v(\cdot)] = \int_0^T \psi(t) f_u[t] v(t) dt + \\ + \int_{M^0} \left| \int_0^T \psi(t, t') f_u[t] v(t) dt \right| dt', \quad (17)$$

а для приращения функционала при $\delta u(\cdot) = sv(\cdot)$, имеем выражение ($s > 0$)

$$F[u(\cdot) + sv(\cdot)] = sD[u(\cdot), v(\cdot)] + o(s) + F[u(\cdot)]. \quad (18)$$

Заметим, что, в отличие от функционала (1), вычисление производной в направлении $-v(\cdot)$, по существу, не требует новых вычислений: достаточно изменить знак у первого слагаемого правой части (17). О практическом использовании формулы (17) можно сказать то же, что и об использовании (10). Для завершения следует исправить неточность, допущенную в формуле (14). Формула будет верна, если множества M^0 , M^- , M^+ заменить множествами M_s^0 , M_s^- , M_s^+ . Строятся они так: положив $C = \max_t |\Phi_x[t] y(t)|$, отнесем t к M_s^0 (M_s^- , M_s^+), если

$$|\Phi[x(t)]| \leq Cs \quad (\Phi[x(t)] < -Cs, \quad \Phi[x(t)] > Cs).$$

Очевидны включения

$$M^0 \subset M_s^0, \quad M_s^- \subset M^-, \quad M_s^+ \subset M^+.$$

Несложные рассуждения позволяют утверждать, что

$$\begin{aligned} \text{mes}(M_s^0/M^0) &\leq \eta(s), \\ \text{mes}(M^+/M_s^+) &\leq \eta(s), \\ \text{mes}(M^-/M_s^-) &\leq \eta(s), \end{aligned}$$

где $\eta(s) \rightarrow 0$ при $s \rightarrow 0$. В самом деле, например, множества M^+/M_s^+ определяются неравенством $Cs > \Phi[x(t)] > 0$ и образуют семейство монотонно убывающих при $s \rightarrow 0$ открытых множеств. Если $\text{mes}(M^+/M_s^+) \geq a > 0$, то существует интервал $[t_1, t_2] \in M^+/M_s^+$ при всех s . Получаем противоречие, так как, с одной стороны, $[t_1, t_2] \notin M^0$, а с другой, $\Phi[x(t)] = 0$ на $[t_1, t_2]$. Учитывая очевидные соотношения

$$\begin{aligned} M_s^0 &= M^0 \cup (M^+/M_s^+) \cup (M^-/M_s^-); \quad M^0 \cap (M^+/M_s^+) = \emptyset; \\ M^0 \cap (M^-/M_s^-) &= \emptyset; \quad M^+ = M_s^+ \cup (M^+/M_s^+); \\ M^- &= M_s^- \cup (M^-/M_s^-), \end{aligned}$$

преобразуем точный вариант (14), упростив обозначения

$$\begin{aligned}
 F[u + sv] = & \int_{M_s^0} |\Phi + s\Phi_x y| dt + \int_{M_s^+} (\Phi + s\Phi_x y) dt - \int_{M_s^-} (\Phi + s\Phi_x y) dt + O(s^2) = \\
 = & s \int_{M^0} |\Phi_x y| dt + \int_{M^+/M_s^+} |\Phi + s\Phi_x y| dt + \int_{M^-/M_s^-} |\Phi + s\Phi_x y| dt + \\
 & + \int_{M^+} (\Phi + s\Phi_x y) dt - \int_{M^+/M_s^+} (\Phi + s\Phi_x y) dt - \\
 & - \int_{M^-} (\Phi + s\Phi_x y) dt + \int_{M^-/M_s^-} (\Phi + s\Phi_x y) dt + O(s^2).
 \end{aligned}$$

В последнем выражении первое, четвертое и шестое слагаемые образуют основную часть формулы (14), а лишние: второе, третье, пятое и седьмое слагаемые оцениваются величинами типа $O(s)\eta(s)$, так как подынтегральные выражения имеют величину $O(s)$, а меры множеств, по которым они интегрируются, не превосходят $\eta(s)$. Таким образом, формула (14) станет верной, если в ней $O(s^2)$ заменить на $o(s)$. Следует подчеркнуть, что при $\text{mes } M^0 = 0$, функционал (2) дифференцируем по Фреше; его производная вычисляется после решения одной задачи (16) для $\psi(t)$ по формуле

$$\frac{\partial F[u(\cdot)]}{\partial u(\cdot)} = f_u^*[t] \psi(t).$$

Заметим, наконец, что проведенный выше анализ функционала (2) использовал гладкость $x(t)$ (при доказательстве того, что $\eta(s) \rightarrow 0$, если $s \rightarrow 0$), однако от функции $y(t)$ требовалась лишь ограниченность. Поэтому обобщение анализа на функционал вида

$$F[u(\cdot)] \equiv \int_0^T |\Phi[x(t), u(t)]| dt$$

потребуется несущественного усложнения, если дополнительно предположить, что вычисление производной осуществляется не на произвольном измеримом управлении, а на кусочно непрерывной функции $u(t)$; на вариацию же управления $sv(t)$ никаких условий (кроме ограниченности) не накладывается. В самом деле, формула (14) в этом случае примет вид

$$F[u(\cdot) + sv(\cdot)] = \int_0^T |\Phi[t] + s\Phi_x[t]y(t) + s\Phi_u[t]v(t)| dt + O(s^2).$$

Так как функция $\Phi[x(t), u(t)]$ — кусочно непрерывна, то дело сводится к проведенному выше анализу на каждом участке непрерывности этой функции.

§ 5. Принцип максимума Л. С. Понтрягина — необходимое условие оптимальности управления

Здесь будет получено необходимое условие, которому удовлетворяет решение следующей задачи оптимального управления: для системы

$$\frac{dx}{dt} = f(x, u), \quad \Gamma(x) = 0, \quad 0 \leq t \leq T, \quad (1)$$

найти управление $u(\cdot)$, минимизирующее функционал F_0 :

$$F_0[u(\cdot)] \rightarrow \min_{u(\cdot)}, \quad (2)$$

при условиях

$$u(t) \in U \quad \text{при всех} \quad t \in [0, T], \quad (3)$$

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m. \quad (4)$$

Все входящие в постановку задачи функционалы предполагаются дифференцируемыми по Фреше (см. § 3), т. е. на любой заданной управлением $u(\cdot)$ траектории $x(t)$ могут быть вычислены (обычно приближенно, но с любой необходимой точностью) значения функционалов $F_i[u(\cdot)]$ и их производные

$$w_i(t) = \frac{\partial F_i[u(\cdot)]}{\partial u(t)}, \quad i = 0, 1, \dots, m. \quad (5)$$

Техника вычисления функций $w_i(t)$ подробно изложена в § 3, здесь подчеркнем лишь, что при этом требуется $(m+1)$ раз решить краевую задачу:

$$\frac{d\psi^{(i)}}{dt} + f_x^*[t]\psi^{(i)} = -Y^{(i)}(t); \quad \Gamma_x^*\psi^{(i)} = 0, \quad i = 0, 1, \dots, m. \quad (6)$$

Для конструирования приближенных методов решения вариационных задач очень важна и интересна и «негативная» формулировка принципа: если условия принципа максимума не имеют места, то исследуемая траектория $\{u(\cdot), x(\cdot)\}$ *) не является оптимальной, в ее окрестности может быть найдена «лучшая» траектория $\{u(\cdot) + \delta u(\cdot), x(\cdot) + \delta x(\cdot)\}$, и нужно уметь ее найти. Поэтому следующий ниже вывод принципа максимума редакционно отличается от общепринятых, имеющих целью лишь «утвердительную» сторону принципа.

Локальный вариант принципа максимума будет получен, если мы ограничимся возмущениями управления $\delta u(\cdot)$, малыми относительно нормы

$$\|\delta u(\cdot)\| \equiv \max_{i=1,2,\dots,r} \max_t |\delta u_i(t)|. \quad (7)$$

*) Под траекторией $\{u(\cdot), x(\cdot)\}$ мы понимаем пару функций $u(t)$ и $x(t)$, удовлетворяющих условиям (1).

Величина $\|\delta u(\cdot)\|$ явно в нижеследующее не входит; малость ее означает, что все величины типа $O(\|\delta u(\cdot)\|^2)$ считаются пренебрежимо малыми. Введем важные в дальнейшем объекты.

Конус U -допустимых вариаций управления K_u . Произвольную вектор-функцию $\delta u(\cdot)$ будем называть U -допустимой в точке $u(\cdot)$, если существует число $s_0 > 0$ такое, что

$$u(t) + s\delta u(t) \in U \quad \text{при всех } t \text{ и } 0 \leq s \leq s_0. \quad (8)$$

Множество таких возмущений управления назовем K_u и выражение $\delta u(\cdot) \in K_u$ будем понимать в смысле (8). Из этого определения

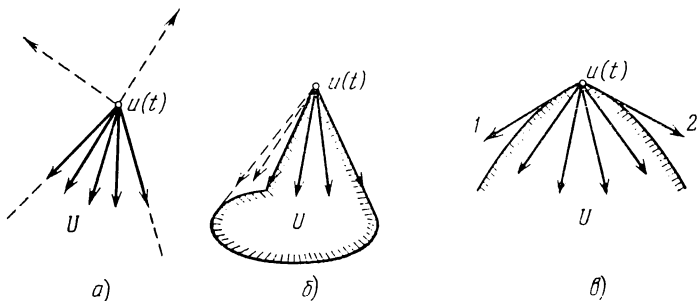


Рис. 3.

следует, что если $\delta u(\cdot) \in K_u$, то и $\lambda \delta u(\cdot) \in K_u$ при любом $\lambda > 0$, следовательно, K_u есть конус в функциональном пространстве.

Фактическое построение конуса K_u в точке $u(\cdot)$ в прикладных задачах обычно очень просто: нужно проанализировать положение $u(t)$ в U при каждом t независимо от остальных значений $u(\cdot)$ и построить конус допустимых вариаций δu в r -мерном пространстве; обозначим этот конус $K(t)$. Поскольку в реальных задачах геометрия области U не очень сложна, это построение будем считать элементарным. Если для каждого t построен свой конус $K(t)$ в r -мерном пространстве, то конус K_u в функциональном пространстве управлений получим как топологическое произведение

$$K_u = \prod_{t \in [0, T]} K(t). \quad (9)$$

Вычислительная реализация конструкции (9) будет рассмотрена ниже; она довольно проста (см. стр. 169). Построение конуса $K(t)$ поясним рис. 3, на котором сплошными линиями изображены U -допустимые направления смещения, а штриховыми — недопустимые.

Изображенные на рис. 3 ситуации заслуживают пояснения. Для области U , б) пунктирными линиями изображены направле-

ния, недопустимые с принятой пока локальной точки зрения. При более общем анализе, основанном на расширении множества вариаций управления за счет конечных вариаций управления на множествах малой меры, такие направления окажутся допустимыми.

Для области U, θ в качестве допустимых изображены направления 1 и 2, касающиеся границы U ; строго говоря, в смысле определения, данного выше, они недопустимы. Однако если определенный выше конус $K(t)$ замкнуть, эти направления войдут в замыкание $\bar{K}(t)$. Эту операцию мы будем считать выполненной. Вопрос о влиянии замыкания на содержательную ценность необходимых условий оптимальности, если он оказывается существенным, требует привлечения второй вариации функционалов задачи.

Множество достижимости. Конус смещений K_F . Каждому управлению $u(\cdot)$ соответствует точка $F[u(\cdot)] = \{F_0[u(\cdot)], \dots, F_m[u(\cdot)]\}$ в пространстве E_{m+1} . Множество точек $F[u(\cdot)]$, порожденных всеми возможными U -допустимыми управлениями, образует множество достижимости D для системы (1) ($D \subset E_{m+1}$). Каждая вариация управления $\delta u(\cdot) \in K_u$ (сдвигает на) вариацию точки $F[u(\cdot)]$ в E_{m+1}

$$\delta F[\delta u(\cdot)] = \int_0^T W(t) \delta u(t) dt. \quad (10)$$

Здесь $W(t)$ — матрица $r \rightarrow m+1$, i -я строка которой является r -мерной вектор-функцией $w_i(t)$, производной функционала $F_i[u(\cdot)]$; $W(t)$ называют *матрицей влияния*; она позволяет вычислить влияние малого возмущения управления на положение изображающей точки $F[u(\cdot) + \delta u(\cdot)]$ в E_{m+1} (вычислить, разумеется, лишь в первом порядке, с точностью до $O(\|\delta u\|^2)$). Таким образом, формула (10) определяет отображение конуса всевозможных вариаций управления K_u в конус смещений K_F ; то, что все δF образуют конус — очевидно: если смещение δF соответствует вариации $\delta u(\cdot) \in K_u$, то смещение $\lambda \delta F$ (где $\lambda > 0$) соответствует вариации $\lambda \delta u(\cdot) \in K_u$. Очень важна для дальнейшего:

Лемма 1. *Замыкание конуса смещений K_F является выпуклым конусом.*

Доказательство состоит в следующем: пусть смещения $\delta F'$ и $\delta F''$ лежат в K_F , т. е. каждое из них порождается своей вариацией $\delta u'(\cdot) \in K_u$, $\delta u''(\cdot) \in K_u$. Лемма будет доказана, если при любых $\alpha, \beta > 0$, $\alpha + \beta = 1$, будет построено смещение $\delta u(\cdot) \in K_u$, причем для соответствующего смещения справедливо тождество:

$$\delta F \equiv \int_0^T W(t) \delta u(t) dt = \alpha \delta F' + \beta \delta F'', \quad (11)$$

Этот факт совершенно очевиден, если все конусы $K(t)$ — выпуклые, следовательно, и K_u — выпуклый конус, а необходимая вариация $\delta u(\cdot)$ строится просто:

$$\delta u(\cdot) = \alpha \delta u'(\cdot) + \beta \delta u''(\cdot) \in K_u.$$

Этот вариант приведен потому, что в прикладных задачах, как правило, область U — выпуклая, следствием чего является выпуклость конуса K_u . Однако в теории оптимального управления K_F оказывается выпуклым конусом и в случае, когда ни один из конусов $K(t)$ не является выпуклым. Установление этого факта является существенным элементом построенной Л. С. Понтягиным и его учениками математической теории оптимального управления. Мы покажем, что для любого сколь угодно малого ε может быть построена вариация $\delta u_\varepsilon(\cdot) \in K_u$, для которой (11) выполнено с точностью до $O(\varepsilon)$. Этим будет установлено, что замыкание K_F является выпуклым конусом, и этого достаточно для дальнейших выводов.

Напомним, что мы условились считать исследуемое управление $u(\cdot)$ кусочно непрерывной функцией. Следствием этого является кусочная непрерывность матрицы влияния $W(t)$. Известно, что при любом сколь угодно малом ε измеримая функция совпадает с некоторой непрерывной функцией всюду, за исключением точек некоторого множества меры ε . Пусть $v'_\varepsilon(\cdot)$ и $v''_\varepsilon(\cdot)$ — соответствующие непрерывные аппроксимации $\delta u'(\cdot)$ и $\delta u''(\cdot)$. Тогда

$$\int_0^T W(t) \delta u'(t) dt = \int_0^T W(t) v'_\varepsilon(t) dt + O(\varepsilon).$$

Разобьем $[0, T]$ на большое число N интервалов точками $0 = t_0 < t_1 < t_2 < \dots < t_N = T$, считая шаг этой сетки равным $O(1/N)$. Пусть точки разрывов $W(t)$ входят в узлы сетки. Каждый интервал (t_i, t_{i+1}) точкой $t_{i+1/2}$ поделим на две части так, что

$$(t_{i+1/2} - t_i) = \alpha(t_{i+1} - t_i), \quad (t_{i+1} - t_{i+1/2}) = \beta(t_{i+1} - t_i).$$

Совокупность левых частей интервалов обозначим M_α , правых — M_β , и определим искомую вариацию

$$\delta u(t) = \begin{cases} \delta u'(t) & \text{при } t \in M_\alpha, \\ \delta u''(t) & \text{при } t \in M_\beta. \end{cases}$$

Очевидно, $\delta u(t) \in K_u$. Вычислим

$$\int_0^T W(t) \delta u(t) dt = \int_{M_\alpha} W(t) \delta u'(t) dt + \int_{M_\beta} W(t) \delta u''(t) dt.$$

Далее,

$$\begin{aligned} \int_0^T W(t) \delta u'(t) dt &= \sum_{i=0}^{N-1} \int_{t_i}^{t_{i+1}} W(t) v'(t) dt + O(\varepsilon) = \\ &= \sum_{i=0}^{N-1} W(t_{i+1/2}) v'(t_{i+1/2}) (t_{i+1} - t_i) + O(\eta(N)) + O(\varepsilon). \end{aligned}$$

Это преобразование сделано на основании непрерывности $W(t) v'(t)$; $\eta(N) \rightarrow 0$ при $N \rightarrow \infty$. Аналогично,

$$\begin{aligned} \int_{M_\alpha} W(t) \delta u'(t) dt &= \sum_{i=0}^{N-1} \int_{t_i}^{t_{i+1/2}} W(t) v'(t) dt + O(\varepsilon) = \\ &= \sum_{i=0}^{N-1} W(t_{i+1/2}) v'(t_{i+1/2}) \alpha(t_{i+1} - t_i) + O(\eta(N)) + O(\varepsilon) = \\ &= \alpha \int_0^T W(t) \delta u'(t) dt + O(\eta(N)) + O(\varepsilon). \end{aligned}$$

Таким образом, доказано, что в любой $O(\varepsilon)$ -окрестности смещения $\alpha \delta F' + \beta \delta F''$ найдется принадлежащее K_F смещение

$$\delta F[\delta u(\cdot)] \equiv \int_0^T W(t) \delta u(t) dt.$$

Лемма доказана, и принцип максимума теперь можно получить, используя важное свойство выпуклых конусов:

Альтернатива для выпуклого конуса: выпуклый конус K_F в конечномерном пространстве E_{m+1} либо совпадает со всем пространством, либо занимает не более полупространства. В последнем случае существует опорный вектор $g \in E_{m+1}$ такой, что

$$(\delta F, g) \leq 0 \quad \text{для всех } \delta F \in K_F.$$

Пусть $u(\cdot)$ — некоторое U -допустимое управление, удовлетворяющее и дополнительным условиям

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m.$$

Такое управление называют *допустимым*; в области достижимости ему соответствует точка $F[u(\cdot)]$, лежащая на оси F_0 ; пусть построен конус смещений K_F (рис. 4). Если управление $u(\cdot)$ — оптимально, конус K_F не должен содержать направления $e = \{-1, 0, 0, \dots, 0\}$. В самом деле, если $e \in K_F$, то существует вариация управления $\delta u(\cdot) \in K_u$ такая, что порожденное ею смещение $\delta F[\delta u(\cdot)] = \{\delta F_0, 0, 0, \dots, 0\}$ и $\delta F_0 < 0$, т. е. значение $F_0[u(\cdot) + \delta u(\cdot)] < F_0[u(\cdot)]$.

а дополнительные условия (в первом порядке по $\|\delta u(\cdot)\|$) не нарушаются; тем самым в окрестности $u(\cdot)$ существует «лучшее» допустимое управление $u(\cdot) + \delta u(\cdot)$. Но выпуклый конус K_F ,

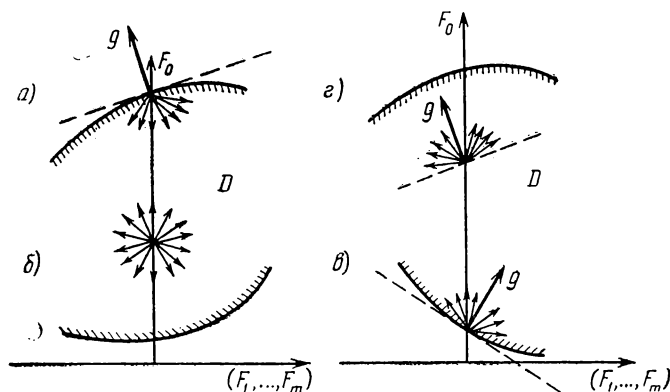


Рис. 4.

не содержащий направления e , не содержит и некоторого полупространства. Другими словами, существует вектор g такой, что

1. $(g, e) = 1 \quad (> 0)$, т. е. $g^0 = -1$,
 2. $(\delta F, g) \leq 0$ для всех $\delta F \in K_F$.
- (12)

Последнее условие следует преобразовать, вводя выражение для

$$\delta F = \int_0^T W(t) \delta u(t) dt.$$

Для всех $\delta u(\cdot) \in K_u$ должно быть

$$0 \geq \left(\int_0^T W(t) \delta u(t) dt, g \right) = \int_0^T (g, W(t) \delta u(t)) dt = \int_0^T (W^*(t) g, \delta u(t)) dt.$$

Так как условие $\delta u(\cdot) \in K_u$ есть «произведение» независимых при разных t условий $\delta u(t) \in K_t$, то из полученного соотношения следует: $(W^*(t) g, \delta u(t)) \leq 0$ для всех $\delta u(t) \in K_t$ и для всех $t \in [0, T]$ (кроме, может быть, множества меры нуль).

Напомним, что i -я строка матрицы $W(t)$ есть функция

$$w^{(i)}(t) = \tilde{w}^{(i)}(t) + f_u^*[t] \psi^{(i)}(t), \quad i = 0, 1, \dots, m,$$

и тогда

$$\begin{aligned} W^*(t)g &= \sum_{i=0}^m g^i \tilde{w}^{(i)}(t) + \sum_{i=0}^m g^i f_u^*[t] \psi^{(i)}(t) = \\ &= \sum_{i=0}^m g^i \tilde{w}^{(i)}(t) + f_u^*[t] \sum_{i=0}^m g^i \psi^{(i)}(t). \end{aligned} \quad (13)$$

Обозначим $\psi(t) = \sum_{i=0}^m g^i \psi^{(i)}(t)$. Так как каждая вектор-функция $\psi^{(i)}(t)$ есть решение уравнения

$$\frac{d\psi^{(i)}}{dt} + f_x^*[t] \psi^{(i)} = -Y^{(i)}(t); \quad \Gamma_x^* \psi^{(i)} = 0,$$

то

$$\frac{d\psi}{dt} + f_x^*[t] \psi = - \sum_{i=0}^m g^i Y^{(i)}(t); \quad \Gamma_x^* \psi = 0. \quad (14)$$

Заметим, что функции $\tilde{w}^{(i)}$ возникли при варьировании функционалов

$$F_i[u(\cdot)] \equiv \int_0^T \Phi^{(i)}[x(t), u(t)] dt$$

и $\tilde{w}^{(i)}(t) = \Phi_u^{(i)}[x(t), u(t)]$, следовательно,

$$\sum_{i=0}^m g^i \tilde{w}^{(i)}(t) = \frac{\partial}{\partial u} \left\{ \sum_{i=0}^m g^i \Phi^{(i)}[x(t), u(t)] \right\}.$$

Кроме того*), $f_u^* \sum_{i=0}^m g^i \psi^{(i)} = \frac{\partial}{\partial u} (f[x(t), u(t)], \psi(t))$. Образует теперь функцию $H[x, u, \psi]$, получившую название *функции Гамильтона*:

$$H(x, u, \psi) \equiv \sum_{i=0}^m g^i \Phi^{(i)}(x, u) + (f(x, u), \psi). \quad (15)$$

В терминах этой функции, определенной с точностью до m неизвестных параметров g^1, g^2, \dots, g^m , необходимое условие оптимальности принимает вид

$$\begin{aligned} H_u[x(t), u(t), \psi(t)] \delta u &\leq 0 \\ \text{для всех } \delta u \in K(t) \text{ и } t \in [0, T]. \end{aligned}$$

) Это следует из выкладок: $(f(u + \delta u), \psi) = (f(u) + f_u \delta u, \psi) = (f, \psi) + (f_u \delta u, \psi) = (f, \psi) + (f_u^ \psi, \delta u)$ и, в соответствии с определением производной, $\frac{\partial}{\partial u} (f(u), \psi) = f_u^* \psi$.

Это означает, что при смещении точки u из $u(t)$ по любому невыходящему из U направлению δu ($\delta u \in K_t$) значение функции $H[x(t), u, \psi(t)]$ разве лишь уменьшается по сравнению с $H[x(t), u(t), \psi(t)]$; иными словами, $H[x(t), u, \psi(t)]$ имеет при каждом t локальный максимум в точке $u = u(t)$. Таким образом, может быть сформулирована:

Теорема 1 (принцип максимума Л. С. Понтрягина). Пусть $u(\cdot)$ — оптимальное управление в задаче (1)–(4). Тогда существует некоторый вектор $g = \{-1, g^1, g^2, \dots, g^m\}$ такой, что определяемая им функция $H[x(t), u, \psi(t)]$ в точке $u = u(t)$ имеет локальный максимум.

(Здесь $\psi(t)$ — решение краевой задачи (14), определенной с точностью до g , H имеет форму (15). Изложенное выше иллюстрирует рис. 4, на котором изображена область достижимости D , возможные положения точки $F[u(\cdot)]$ в ней, конусы смещений K_F и векторы g . Обсудим возможные варианты:

а) существует вектор g , удовлетворяющий условию (12,2), но условие нормировки (12,1) не выполнено: $(g, e) < 0$, $e \in K_F$, $u(\cdot)$ не оптимально;

б) вектора g не существует, K_F — все пространство, $e \in K_F$, $u(\cdot)$ не оптимально;

в) вектор g удовлетворяет условиям (12), $e \notin K_F$, $u(\cdot)$ — оптимально;

г) существует g , удовлетворяющий (12), однако $u(\cdot)$ — не оптимально. Эта ситуация соответствует, например, локальному минимуму задачи или «точке перегиба».

Негативная формулировка принципа максимума. Теорема 2. Пусть вектора g , фигурирующего в теореме 1, не существует, т. е. $e \in K_F$; более того, пусть e принадлежит внутренности K_F . Тогда управление $u(\cdot)$ не оптимально, в окрестности $u(\cdot)$ существует лучшее управление $u(\cdot) + \delta u(\cdot)$, для которого

- 1) $F_0[u(\cdot) + \delta u(\cdot)] < F_0[u(\cdot)],$
- 2) $u(t) + \delta u(t) \in U$ при всех $t,$
- 3) $F_i[u(\cdot) + \delta u(\cdot)] = 0, \quad i = 1, 2, \dots, m.$

Эта теорема нуждается в доказательстве лишь в силу нелинейности задачи и необходимости учесть влияние малых величин порядка $O(\|\delta u\|^2)$. Мы не будем приводить ее полного и строгого

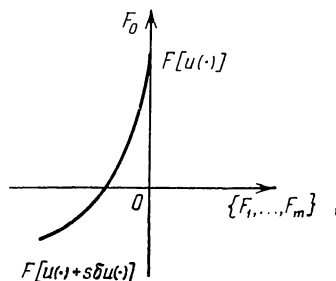


Рис. 5.

доказательства, но лишь наметим его основные моменты. На рис. 5 схематически изображена исследуемая ситуация.

Пусть $\delta u(\cdot)$ — вариация управления, отображаемая в $e \in K_F$:

$$e = \int_0^T W(t) \delta u(t) dt.$$

Рассмотрим луч U -допустимых смещений в пространстве управлений: $u(\cdot) + s\delta u(\cdot)$, $s \geq 0$. В линейном приближении движение по этому лучу (увеличение s от нуля) сопровождается движением точки F по оси F_0 вниз:

$$F[u(\cdot) + s\delta u(\cdot)] = F[u(\cdot)] + se,$$

реальное же движение точки F с учетом нелинейности задачи имеет вид (см. рис. 5)

$$F[u(\cdot) + s\delta u(\cdot)] = F[u(\cdot)] + se + O(s^2).$$

Выберем теперь в конусе $K_F(m+1)$ элемент $\{\delta F^{(0)}, \delta F^{(1)}, \dots, \delta F^{(m)}\}$ так, чтобы направление e лежало строго внутри $(m+1)$ -гранного конуса, натянутого на $\delta F^{(i)}$. Это возможно, так как e , по предположению, лежит строго внутри K_F . Таким образом,

$$e = \sum_{i=0}^m \alpha_i \delta F^{(i)}, \quad \alpha_i \geq 0.$$

Каждому $\delta F^{(i)}$ соответствует $\delta u^{(i)}(\cdot) \in K_u$ (или $\delta u(\cdot)$), порождающая сколь угодно близкое к $\delta F^{(i)}$ смещение; мы не станем осложнять изложение учетом этой поправки, так как она легко включается в дальнейшие оценки. Образовав «смесь» с весами α_i *) вариаций $\delta u^{(i)}(\cdot)$, получим вариацию $\delta u(\cdot)$, порождающую в первом порядке смещение

$$\delta F = s \sum_{i=1}^m \alpha_i \delta F^{(i)} = se.$$

Однако реальное смещение, как уже было отмечено, имеет вид

$$F[u(\cdot) + s\delta u(\cdot)] = F[u(\cdot)] + se + s^2 r^{(1)}, \quad r^{(1)} = O(1).$$

При достаточно малом s точка $se - s^2 r^{(1)}$ лежит внутри построенного выше $(m+1)$ -гранного конуса, поэтому можно ввести коррекцию в числа α_i , подобрав их так, чтобы в линейном приближении выполнялось равенство

$$\sum_{i=1}^m \tilde{\alpha}_i \delta F^{(i)} = e - sr^{(1)}.$$

*) Например, разбив $[0, T]$ на N интервалов и поделив каждый на пропорциональные α_i части.

При этом $\tilde{\alpha}_i = \alpha_i + \beta_i$, $|\beta_i| = O(s)$, $\tilde{\alpha}_i \geq 0$. Пусть $\delta \tilde{u}(\cdot)$ — новая вариация, соответствующая смещению

$$\delta \tilde{F} = \sum_{i=0}^m \tilde{\alpha}_i \delta F^{(i)}, \quad \delta \tilde{u}(\cdot) = \delta u(\cdot) + O(s).$$

Тогда

$$F[u(\cdot) + s \delta \tilde{u}(\cdot)] = F[u(\cdot)] + [se - s^2 r^{(1)}] + s^2 \tilde{r}^{(1)}.$$

Так как квадратичный член $s^2 \tilde{r}^{(1)}$ определяется величиной возмущения, то для возмущений $\delta u(\cdot)$ и $\delta \tilde{u}(\cdot)$, отличающихся на $O(s)$, квадратичные члены совпадают с точностью до малых высшего порядка, т. е. $s^2 r^{(1)} = s^2 \tilde{r}^{(1)} + O(s^3)$. Итак,

$$F[u(\cdot) + s \delta \tilde{u}(\cdot)] = F[u(\cdot)] + se + O(s^3).$$

Таким же образом можно ликвидировать ошибку $O(s^3)$, заменив ее на $O(s^4)$ и т. д. Этим мы и ограничимся здесь, не рассматривая вопросов о построении предельной вариации $\delta u(\cdot)$, удовлетворяющей условиям 1, 2, 3 теоремы 2, тем более, что без дополнительных предположений измеримый предел $\delta u(\cdot)$ может и не существовать. Возникающие здесь тонкие вопросы предельных переходов обсуждаются в § 10. Забегая вперед, отметим, что существование предельной вариации $\delta u(\cdot)$, в сущности, и не нужно. Важно то, что соответствующие построенной последовательности управлений $u(\cdot) + \delta u^{(k)}(\cdot)$, $k=1, 2, \dots$, траектории $x^{(k)}(\cdot)$ образуют компактное семейство, и существует предельная траектория $\lim_{k \rightarrow \infty} x^{(k)}(\cdot) = x^*(\cdot)$, для которой

$$\lim_{k \rightarrow \infty} F_0[u(\cdot) + \delta u^{(k)}(\cdot)] \leq F_0[u],$$

$$\lim_{k \rightarrow \infty} F_i[u(\cdot) + \delta u^{(k)}(\cdot)] = 0; \quad i = 1, 2, \dots, m.$$

А такую ситуацию естественно трактовать как неоптимальность траектории $\{u(\cdot), x(\cdot)\}$, даже если предельная траектория $x^*(t)$ и не является решением краевой задачи (1) с каким-то измеримым управлением $u^*(t)$.

В заключение сделаем два замечания.

Первое имеет формальный характер и относится к небольшой нестрогости, допущенной при доказательстве того, что \bar{K}_F есть выпуклый конус. Это доказательство в действительности нужно было бы провести в такой форме: пусть $\delta F'$ — такое смещение, что для любого сколь угодно малого $\delta > 0$ найдется вариация $\delta u'(\cdot) \in K_u$, причём

$$\left\| \delta F' - \int_0^T W(t) \delta u'(t) dt \right\| \leq \delta,$$

т. е. $\delta F'$ принадлежит замыканию K_F ; $\delta F''$ следует предположить таким же смещением. Тогда для $\delta F = \alpha \delta F' + (1 - \alpha) \delta F''$ и любого $\varepsilon > 0$ нужно построить $\delta u(\cdot) \in K_u$ такое, что

$$\left\| \delta F - \int_0^T W(t) \delta u(t) dt \right\| \leq \varepsilon.$$

Читатель без труда внесет соответствующие изменения в доказательство.

Второе замечание носит более содержательный характер и касается возможности сразу же из локального варианта принципа максимума получить глобальную (по $u \in U$) формулировку. Разумеется, это возможно лишь при определенных предположениях о входящих в постановку задачи функциях, однако эти предположения оправдываются в большом числе прикладных задач. Для простоты изложения предположим, что все функционалы задачи имеют форму $F[u(\cdot)] \equiv \Phi[x(t')]$. Общность от этого не теряется, так как к такой форме легко сводятся и задачи с функционалами

$F[u(\cdot)] \equiv \int_0^T \Phi[x, u] dt$. Это достигается стандартным приемом рас-

ширения фазового пространства системы: каждый такой функционал порождает добавочную компоненту x с уравнением для нее типа

$$\frac{dx^{n+1}}{dt} = \Phi[x, u], \quad x^{n+1}(0) = 0,$$

после чего $F[u(\cdot)] \equiv \int_0^T \Phi[x, u] dt = x^{n+1}(T)$.

Введем важное для дальнейшего понятие. *Вектограммой* управляемой системы $\dot{x} = f(x, u)$ в точке x называется множество точек $f(x, u)$ в E_n , порождаемое всеми значениями $u \in U$. Вектограмму будем в дальнейшем обозначать $f(x, U)$; она описывает, в некотором смысле, технические возможности управляемой системы: за малое время τ система из точки x может попасть лишь в точки множества $x + \tau f(x, U)$ (с точностью до $O(\tau^2)$).

Т е о р е м а 3. Пусть вектограмма системы $\dot{x} = f(x, u)$ выпукла, а все функционалы задачи имеют вид $\Phi[x(t')]$ (или являются функциями таких функционалов). Тогда на оптимальной траектории $\{u(\cdot), x(\cdot)\}$ может быть определено решение $\phi(t)$ системы уравнений вида (14) такое, что имеет место соотношение

$$H[x(t), \phi(t), u(t)] = \max_{u \in U} H[x(t), \phi(t), u]. \quad (16)$$

Доказательство. В рассматриваемом случае

$$\begin{aligned} H[x, \psi, u] &= (f(x, u), \psi), \\ \max_{u \in U} (f(x, u), \psi) &= \max_{z \in f(x, U)} (z, \psi). \end{aligned} \quad (17)$$

Но при любом заданном ψ линейная форма (z, ψ) на выпуклом множестве $f(x, U)$ точек локального максимума, не совпадающих с точками глобального максимума, не имеет. Другими словами, точка u^* , являющаяся точкой локального максимума $(f(x, u), \psi)$, в то же время является и точкой глобального максимума. В следующем параграфе глобальная формулировка (16) будет доказана без предположения о выпуклости $f(x, U)$.

Дискретный принцип максимума. В последние годы появилось большое число работ*), в которых необходимое условие оптимальности формулируется для дискретных управляемых систем. Состояние такой системы описывается не функцией $x(t)$, а дискретным набором $x_0, x_1, \dots, x_n, \dots, x_k$, а эволюция — уравнением

$$x_{k+1} = f(x_k, u_k), \quad x_0 — \text{задано}. \quad (18)$$

Необходимость изучения задачи (18) аргументируется, в частности, потребностями численного решения задач оптимального управления: ведь численные методы имеют дело не с дифференциальными уравнениями, а с их разностными аппроксимациями

$$x_{k+1} = x_k + dt f(x_k, u_k), \quad (19)$$

поэтому исследование задач типа (18), (19), приводящее к дискретному принципу максимума, имеет большое значение для практики. Выше была приведена характерная схема рассуждений. Однако в настоящей книге, имеющей практические вычисления в качестве основной цели, мы нигде исследованиями по дискретному принципу максимума пользоваться не будем, хотя во всех рассматриваемых примерах под решениями дифференциальных уравнений понимаются именно дискретные последовательности, полученные по формулам типа (19). Это связано с тем, что автор не смог извлечь никаких реальных рекомендаций, которые следовали бы из отличия дискретного принципа максимума от принципа максимума для дифференциальных уравнений и которые нужно было бы использовать в практических вычислениях. Чтобы придать этому высказыванию более четкий смысл, рассмотрим единственное известное автору реальное практическое следствие дискретного принципа максимума. Как известно, при выводе принципа максимума для дифференциальных уравнений используются следующие основные объекты:

*) См. [69], [57].

I. Исходное уравнение $\dot{x} = f(x, u)$.

II. Уравнение в вариациях $\delta\dot{x} = f_x\delta x + f_u\delta u$.

III. Тождество Лагранжа

$$\int_0^T \{\psi(\delta\dot{x} - f_x\delta x) + \delta x(\dot{\psi} + f_x^*\psi)\} dt = \psi\delta x|_0^T.$$

IV. Сопряженное уравнение $\dot{\psi} + f_x^*\psi = Y(t)$.

Дискретный принцип максимума получается почти по такой же схеме, но вместо дифференциальных уравнений в выкладках участвуют их разностные аппроксимации. И вот здесь появляется упомянутое реальное следствие дискретной теории: разностное уравнение для сопряженного уравнения является следствием того или иного выбора аппроксимаций для прямого уравнения и для интеграла в тождестве Лагранжа. Разностная аппроксимация уравнения в вариациях также однозначно определяется выбором аппроксимации исходного уравнения, но это не так важно, так как в вычислительных методах обычно это уравнение не интегрируется. Эту аппроксимацию сопряженного уравнения мы будем называть *согласованной* с аппроксимациями исходного уравнения и интеграла в том смысле, что для конечно-разностных решений δx и ψ , полученных по согласованным аппроксимациям соответствующих уравнений, алгебраически точно выполняется тождество Лагранжа (тоже в соответствующей аппроксимации). Это и есть то единственное практическое следствие, которое автор смог извлечь из теории дискретного принципа максимума и которого в своих вычислениях никогда не использовал ни в явной, ни в неявной формах. Автор всегда выбирал для исходного и сопряженного уравнений независимые аппроксимации, причем сопряженное обычно интегрировалось более грубо, с большим шагом по времени. Дело в том, что использование согласованной аппроксимации связано с определенными техническими неудобствами, необходимость преодоления которых не очевидна. Во всяком случае, автору неизвестны трудности численного решения задач оптимального управления, которые можно было бы преодолеть, используя согласованную аппроксимацию. Чтобы и здесь быть более конкретным, можно все же указать на некоторое следствие использования согласованной аппроксимации. Речь идет о получении минимума функционала с большим числом знаков. Используя для вычисления функциональной производной функцию ψ , найденную по произвольной аппроксимации сопряженного уравнения, мы, разумеется, находим не точную производную, а лишь приближенную, искаженную влиянием ошибок аппроксимации. Поэтому получить минимум с очень большой точностью не удастся; начиная с некоторого этапа минимизации (например, методом градиента в функциональном пространстве) мы будем в этом случае

двигаться, в сущности, по случайному, определяемому ошибками аппроксимации, направлению. Используя согласованную на разностном уровне аппроксимацию, мы вычисляем точную (не считая ошибок округления) производную функционала (точнее, его разностной аппроксимации; кстати, если функционал имел вид $\int \Phi(x, u)dt$, то аппроксимация тождества Лагранжа индуцируется аппроксимацией этого интеграла). Это дает основание надеяться на получение минимума с большим числом знаков. Однако такое преимущество представляется сомнительным: ведь на этом этапе мы уточняем не значение исходного функционала непрерывной задачи, а лишь те его знаки, которые определяются ошибкой выбранного способа его разностной аппроксимации. В § 26 приведен пример (решение задачи о брахистохроне), в котором все же была использована согласованная аппроксимация.

§ 6. Принцип максимума. Конечные вариации управления на множестве малой меры

В теории оптимального управления, кроме рассмотренных уже возмущений управления малыми функциями, оказывается возможным и другой класс возмущений, при которых функция $u(t)$ изменяется на конечную величину, но не на всем интервале $[0, T]$, а на некотором его подмножестве, мера которого мала и является в данном случае величиной первого порядка малости.

Итак, рассматриваются система с управлением

$$\frac{dx}{dt} = f(x, u); \quad \Gamma(x) = 0; \quad u(t) \in U, \quad (1)$$

и определенный на траектории $\{u(\cdot), x(\cdot)\}$ функционал

$$F[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt$$

или $F[u(\cdot)] \equiv \Phi[x(t')]$, t' — заданная точка. Пусть $v(t)$ — произвольная U -допустимая функция, а M — некоторое подмножество $[0, T]$, мера которого $\mu = \text{mes } M$ мала *). Рассмотрим возмущенное управление

$$u^*(t) = \begin{cases} v(t) & \text{при } t \in M, \\ u(t) & \text{при } t \notin M. \end{cases} \quad (2)$$

*) Можно представлять M как совокупность непересекающихся интервалов; суммарная длина их равна $\mu \ll T$, расположение их на $[0, T]$ произвольно.

1. Уравнение в вариациях. Прежде всего следует доказать, что такое управление $u^*(t)$ определяет фазовую траекторию $x^*(t)$, близкую к $x(t)$, а именно:

$$\|x^*(t) - x(t)\| = O(\mu) \text{ при всех } t \in [0, T], \quad (3)$$

после чего будет получено и уравнение в вариациях для $\delta x(t) = x^*(t) - x(t)$. Мы проведем типичное доказательство для частного (но очень распространенного) случая задачи Коши, когда условия $\Gamma(x) = 0$ имеют вид $x(0) - X_0 = 0$.

Теорема 1. Если управление $u(t)$ возмущено на множестве M малой меры μ , то соответствующее возмущение фазовой траектории системы имеет оценку $\|\delta x(t)\| = O(\mu)$ при всех t и удовлетворяет уравнению в вариациях

$$\frac{d\delta x}{dt} = f_x[x] \delta x + f[x(t), u^*(t)] - f[x(t), u(t)] + O(\mu^2). \quad (4)$$

Доказательство. Имеем уравнения для x^* и x :

$$\frac{dx^*}{dt} = f(x^*, u^*); \quad \frac{dx}{dt} = f(x, u); \quad x(0) = x^*(0) = X_0. \quad (5)$$

Разобьем $[0, T]$ на большое число N равных отрезков точками $0 = t_0 < t_1 < t_2 < \dots < t_N = T$, $t_n = n\tau$, $\tau = T/N$, и обозначим $M_{n+1/2} = M \cap [t_n, t_{n+1}]$, $\mu_{n+1/2} = \text{mes } M_{n+1/2}$; очевидно,

$$M = \bigcup_{n=0}^{N-1} M_{n+1/2}, \quad \mu = \sum_{n=0}^{N-1} \mu_{n+1/2},$$

Рассмотрим и оценим последовательное накопление расхождения между $x^*(t)$ и $x(t)$. Обозначим $x_n^* = x^*(t_n)$, $x_n = x(t_n)$. Тогда

$$\begin{aligned} x_{n+1} &= x_n + \int_{t_n}^{t_{n+1}} f[x(t), u(t)] dt = \\ &= x_n + \int_{t_n}^{t_n+\tau} f[x_n, u(t)] dt + \int_{t_n}^{t_n+\tau} \{f[x(t), u(t)] - f[x_n, u(t)]\} dt = \\ &= x_n + \int_{t_n}^{t_{n+1}} f[x_n, u(t)] dt + O(\tau^2). \end{aligned}$$

Точно так же $x_{n+1}^* = x_n^* + \int_{t_n}^{t_{n+1}} f[x_n^*, u^*(t)] dt + O(\tau^2)$. Теперь

$$\|x_{n+1}^* - x_{n+1}\| \leq \|x_n^* - x_n\| + \left\| \int_{t_n}^{t_{n+1}} \{f[x_n^*, u^*(t)] - f[x_n, u(t)]\} dt \right\| + O(\tau^2).$$

Оценим отдельно

$$\begin{aligned} \int_{t_{n+1}}^{t_{n+1}} \{f[x_n^*, u^*(t)] - f[x_n, u(t)]\} dt &= \int_{t_{n+1}}^{t_{n+1}} \{f[x_n^*, v(t)] - f[x_n, u(t)]\} dt + \\ &+ \int_{[t_{n+1}, t_{n+1}]/M_{n+1/2}}^{M_{n+1/2}} \{f[x_n^*, u(t)] - f[x_n, u(t)]\} dt. \end{aligned}$$

Для оценки интеграла по $M_{n+1/2}$ воспользуемся ограниченностью $\|f(x, u)\| \leq C_0$ при всех x и u , а для оценки интеграла по $[t_n, t_{n+1}]/M_{n+1/2}$ — условием Липшица

$$\|f(x^*, u) - f(x, u)\| \leq C_1 \|x^* - x\|.$$

Получим оценку

$$\left\| \int_{t_n}^{t_{n+1}} \{f[x_n^*, u^*(t)] - f[x_n, u(t)]\} dt \right\| \leq 2C_0 \mu_{n+1/2} + \tau C_1 \|x_n^* - x_n\|.$$

Итак,

$$\|x_{n+1}^* - x_{n+1}\| \leq \|x_n^* - x_n\| (1 + C_1 \tau) + 2C_0 \mu_{n+1/2} + O(\tau^2).$$

Эта рекуррентная оценка стандартным способом приводит к следующей:

$$\begin{aligned} \|x_n^* - x_n\| &\leq (1 + C_1 \tau)^n \|x_0^* - x_0\| + 2C_0 \sum_{k=0}^{n-1} \mu_{k+1/2} (1 + C_1 \tau)^{n-k} + \\ &+ O(\tau^2) \sum_{k=0}^{n-1} (1 + C_1 \tau)^{n-k} \leq 2C_0 e^{C_1 \tau} \sum_{k=0}^N \mu_{k+1/2} + O(\tau^2) \frac{1}{C_1 \tau} \leq \\ &\leq 2C_0 e^{C_1 \tau} \mu + O(\tau). \end{aligned}$$

Этим и заканчивается доказательство первого утверждения теоремы. Что касается уравнения в вариациях для $\delta x(t) = x^*(t) - x(t)$, то оно получается просто: вычитая уравнение для x из уравнения для x^* , производя в $f(x^*, u^*)$ замену $x^* = x + \delta x$, и разлагая в ряд по δx , получим

$$\begin{aligned} \frac{d \delta x}{dt} &= f(x + \delta x, u^*) - f(x, u) = \\ &= f(x, u^*) + f_x(x, u^*) \delta x - f(x, u) + O(\|\delta x\|^2) = \\ &= f_x[x(t), u(t)] \delta x + \{f[x(t), u^*(t)] - f[x(t), u(t)]\} + \\ &+ O(\mu^2) + \{f_x[x(t), u^*(t)] - f_x[x(t), u(t)]\} \delta x. \quad (6) \end{aligned}$$

Последнее слагаемое имеет величину $O(\mu)$ на M и равно нулю вне M ; мы включим его формально в $O(\mu^2)$, имея в виду следующее: решение уравнения в вариациях

$$\frac{d\delta x}{dt} = f_x[t]\delta x + \{f[x(t), u^*(t)] - f[x(t), u(t)]\}, \quad \delta x(0) = 0 \quad (7)$$

отличается от точной разности $x^*(t) - x(t)$, удовлетворяющей уравнению (6), на величину $O(\mu^2)$. Теорема доказана.

2. Вариация функционала. Теперь следует вычислить первую вариацию функционала для возмущений подобного рода. Ограничимся здесь конструкцией $\int \Phi(x, u) dt$, так как для $\Phi[x(t')]$ вывод будет еще проще. Итак,

$$\begin{aligned} & \int_0^T \{\Phi[x^*(t), u^*(t)] - \Phi[x(t), u(t)]\} dt = \\ & = \int_0^T \{\Phi[x(t) + \delta x(t), u^*(t)] - \Phi[x(t), u(t)]\} dt = \\ & = \int_0^T \{\Phi(x, u^*) - \Phi(x, u) + \Phi_x(x, u)\delta x + [\Phi_x(x, u^*) - \Phi_x(x, u)]\delta x\} dt = \\ & = \int_M \{\Phi[x(t), v(t)] - \Phi[x(t), u(t)]\} dt + \int_0^T \Phi_x[t]\delta x(t) dt + \\ & \quad + O(\mu^2) + \int_M \{\Phi_x[x(t), v(t)] - \Phi_x[x(t), u(t)]\}\delta x dt. \end{aligned}$$

Последнее слагаемое есть тоже $O(\mu^2)$, и можно написать формулу $F[u^*(\cdot)] - F[u(\cdot)] =$

$$= \int_M \{\Phi[x(t), v(t)] - \Phi[x(t), u(t)]\} dt + \int_0^T \Phi_x[t]\delta x(t) dt + O(\mu^2).$$

Дальнейшее исключение $\delta x(t)$ при помощи уравнения в вариациях основано на стандартном приеме: определив $\psi(t)$, как решение краевой задачи

$$\frac{d\psi}{dt} + f_x^*[t]\psi = -\Phi_x[t], \quad \Gamma_x^*\psi = 0, \quad (8)$$

и исключая в тождестве Лагранжа (см. § 3) $\frac{d\delta x}{dt} - f_x\delta x = f(x, u^*) - f(x, u)$, получим

$$\begin{aligned} \int_0^T \Phi_x[t] \delta x(t) dt &= \int_0^T \psi(t) \{f[x(t), u^*(t)] - f[x(t), u(t)]\} dt = \\ &= \int_M \psi(t) \{f[x(t), v(t)] - f[x(t), u(t)]\} dt, \end{aligned}$$

и окончательно

$$\begin{aligned} \delta F &= F[u^*(\cdot)] - F[u(\cdot)] = \int_M \{\Phi[x(t), v(t)] - \Phi[x(t), u(t)]\} dt + \\ &+ \int_M \psi(t) \{f[x(t), v(t)] - f[x(t), u(t)]\} dt + O(\mu^2). \quad (9) \end{aligned}$$

3. Принцип максимума. Пусть для системы (1) определены дифференцируемые функционалы $F_i[u(\cdot)], i=0, 1, \dots, m$, и ставится стандартная задача нахождения $\min_{u(\cdot)} F_0[u(\cdot)]$ при условиях $F_i[u(\cdot)] = 0, i=1, 2, \dots, m$.

Пусть исследуется невозмущенная траектория $\{u(\cdot), x(\cdot)\}$; на ней вычислены функционалы $F_i, F_1=F_2=\dots=F_m=0$, и найдены решения $\psi^{(i)}(t), i=0, \dots, m$, краевых задач типа (8). Дальнейшие построения, приводящие к принципу максимума, аналогичны построениям § 5, однако технически несколько отличаются. Мы не будем воспроизводить доказательство со всеми необходимыми тонкостями чисто математического характера, однако в следующем ниже наброске доказательства принципа максимума эти тонкости будут по возможности четко указаны.

«Конус» вариаций управления K_u в данном случае есть множество объектов, состоящих из произвольной U -допустимой функции $v(t)$ и измеримого множества M малой меры $\text{mes } M = \mu$. Формально можно пользоваться обозначением $\{v(\cdot), M\} \in K_u$. Эта совокупность комплексов $\{v(\cdot), M\}$, строго говоря, не есть конус; тем не менее в некотором условном смысле слова можно говорить о конусе: если $\{v(\cdot), M\} \in K_u$, то для любого $\lambda \in [0, 1]$ можно определить объект $\{v(\cdot), M_\lambda\} \in K_u$ так, что порождаемые им вариации функционалов связаны соотношением

$$\delta F[v(\cdot), M_\lambda] = \lambda \delta F[v(\cdot), M] + \varepsilon, \quad (10)$$

где ε — сколь угодно мало (точнее, следовало бы обозначать M_λ символом $M(\lambda, \varepsilon)$). Построить множество $M(\lambda, \varepsilon)$ можно, например, так: разбить $[0, T]$ на большое число N интервалов (t_i, t_{i+1}) , каждый (t_i, t_{i+1}) разбить на две части — левую, длиной $\lambda(t_{i+1} - t_i)$, и правую, длиной $(1 - \lambda)(t_{i+1} - t_i)$. Пересечение совокупности левых частей с M и образует при достаточно большом числе $N > N(\varepsilon)$ нужное множество $M(\lambda, \varepsilon)$, обладающее свойством (10). Для дальнейшего же нет необходимости в том, чтобы K_u было ко-

нусом. Важно другое — чтобы совокупность порождаемых K_u вариаций δF обладала характерными чертами выпуклого конуса.

Теперь рассмотрим конус смещений K_F . Каждое возмущение управления $\{v(\cdot), M\}$ порождает смещение δF значений функционалов в соответствии с главным членом (9):

$$\delta F; [v(\cdot), M] = \int_M \{\Phi^{(i)}[x(t), v(t)] - \Phi^{(i)}[x(t), u(t)]\} dt + \\ + \int_M \psi(t) \{f[x(t), v(t)] - f[x(t), u(t)]\} dt. \quad (11)$$

Основной момент доказательства — это проверка того, что для любых возмущений $\{v'(\cdot), M'\}$ и $\{v''(\cdot), M''\}$ и любого числа $\alpha \in [0, 1]$ можно построить такое возмущение $\{v(\cdot), M\}$, что порожденные этими возмущениями смещения $\delta F'$, $\delta F''$, δF будут связаны соотношением

$$\delta F[v(\cdot), M] = \alpha \delta F[v'(\cdot), M'] + (1 - \alpha) \delta F[v''(\cdot), M''] + \varepsilon$$

со сколь угодно малой величиной $\|\varepsilon\|$. Разумеется, $v(\cdot)$ и M зависят от этого ε . Конструируется M , например, так: $[0, T]$ разбивается на N интервалов, каждый интервал делится на две части, пропорциональные α и $(1 - \alpha)$ соответственно; пересечение M' с совокупностью левых частей маленьких интервалов образует множество M_1 , пересечение M с совокупностью правых частей — множество M_2 ; далее, $M = M_1 \cup M_2$, а $v(\cdot)$, очевидно, есть

$$v(t) = \begin{cases} v'(t) & \text{при } t \in M_1, \\ v''(t) & \text{при } t \in M_2. \end{cases}$$

Малость ε достигается за счет достаточно большого числа N ; при $N \rightarrow \infty$

$$\text{mes } M_1 \rightarrow \alpha \text{ mes } M', \quad \text{mes } M_2 \rightarrow (1 - \alpha) \text{ mes } M''.$$

Таким образом, замыкание K_F оказывается выпуклым конусом и, предположив оптимальность исследуемой траектории $\{u(\cdot), x(\cdot)\}$, так же, как в § 5, получаем существование вектора $g = \{-1, g^1, g^2, \dots, g^m\}$, для которого

$$(g, \delta F) \leq 0 \quad \text{при всех } \delta F \in K_F. \quad (12)$$

Используя формулу (11) для δF и вводя функцию

$$H(x, \psi, u) \equiv \sum_{i=0}^m g^i \Phi^{(i)}(x, u) + \left(\sum_{i=0}^m g^i \psi^{(i)}(t), f(x, u) \right), \quad (13)$$

получим для любых M и $v(\cdot)$ условие (12) в форме

$$\int_M \{H[x(t), \psi(t), v(t)] - H[x(t), \psi(t), u(t)]\} dt \leq 0. \quad (14)$$

Так как множество сколь угодно малой меры может быть расположено всюду плотно на $[0, T]$, то из (14) следует

$$H[x(t), \psi(t), u(t)] = \max_{v \in U} H[x(t), \psi(t), v] \quad (15)$$

при почти всех $t \in [0, T]$.

Напомним, что $\psi(t)$ — решение краевой задачи, формулировка которой содержит неопределенные параметры g^1, \dots, g^m

$$\frac{d\psi}{dt} + f_x[t]\psi = - \sum_{i=0}^m g^i Y^{(i)}(t); \quad \Gamma_x^* \psi = 0. \quad (16)$$

§ 7. Некоторые обобщения задачи оптимального управления

Выше мы ограничились анализом сравнительно простой формы задачи оптимального управления, хотя рассмотреть более общий класс задачи было бы совсем нетрудно. Многие обобщения, по существу, не требуют привлечения новых идей и приводят к затруднениям, в основном, в связи с более громоздкими обозначениями. Такого сорта вариации задачи оптимального управления мы и рассмотрим в настоящем параграфе, ограничиваясь лишь дифференцированием функционалов, входящих в их постановку. Для рассматриваемых здесь обобщений этого достаточно.

1. Задача с параметрами. Пусть в постановку вариационной задачи входит набор параметров $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_p\}$, которые не фиксированы, но должны определяться наряду с управлением $u(\cdot)$ и из тех же соображений. Такую задачу формально можно записать в таком виде:

движение системы определяется уравнением

$$\frac{dx}{dt} = f(x, u, \alpha); \quad \Gamma(x, \alpha) = 0; \quad u(t) \in U; \quad (1)$$

на каждой траектории $\{u(\cdot), x(\cdot), \alpha\}$ определены функционалы типа, например:

$$\begin{aligned} 1) \quad F[u(\cdot), \alpha] &\equiv \int_0^T \Phi[x(t), u(t), \alpha] dt. \\ 2) \quad F[u(\cdot), \alpha] &\equiv \Phi[x(t'), \alpha], \\ 3) \quad F[u(\cdot), \alpha] &\equiv \max_t \Phi[x(t), \alpha]. \end{aligned} \quad (2)$$

В остальном задача ставится обычным образом. В связи с этим полезно ввести

О п р е д е л е н и е. Управлением в широком смысле слова будем называть совокупность функций и параметров, задание

которых однозначно определяет фазовую траекторию управляемой системы и, следовательно, значения входящих в постановку задачи функционалов (которые теперь следует обозначать $F[u(\cdot), \alpha]$). Будем считать, что в задаче (1), (2) комплекс $\{u(\cdot), \alpha\}$ является именно таким управлением, задание которого превращает (1) в краевую задачу, имеющую единственное решение. Исследование этой задачи в точности следует схеме, изложенной в § 5; здесь мы повторим основные этапы этой схемы с соответствующими изменениями.

1. Уравнение в вариациях имеет очевидную форму:

$$\frac{d\delta x}{dt} - f_x[t]\delta x = f_u[t]\delta u + f_\alpha[t]\delta\alpha; \quad \Gamma_x\delta x + \Gamma_\alpha\delta\alpha = 0. \quad (3)$$

2. Прямая вариация функционала:

$$1) \delta F[\delta u(\cdot), \delta\alpha] = \int_0^T \Phi_u[t]\delta u(t)dt + \int_0^T \Phi_x\delta xdt + \int_0^T \Phi_\alpha[t]dt\delta\alpha,$$

$$2) \delta F[\delta u(\cdot), \delta\alpha] = \Phi_x[x(t'), \alpha]\delta x(t') + \Phi_\alpha[x(t'), \alpha]\delta\alpha,$$

$$3) \delta F[\delta u(\cdot), \delta\alpha] = \max_{t \in M} \{\Phi_x[t]\delta x(t) + \Phi_\alpha[x(t), \alpha]\delta\alpha\}.$$

3. Исключение вариаций «зависимого» переменного $\delta x(t)$ из полученных выражений для δF . Для этого используется тождество Лагранжа в обычной форме:

$$\int_0^T \left\{ \psi \left(\frac{d}{dt} - f_x \right) \delta x - \delta x \left(\frac{d}{dt} - f_x \right)^* \psi \right\} dt = \psi \delta x|_0^T. \quad (4)$$

Заменив $\left(\frac{d}{dt} - f_x \right) \delta x = f_u \delta u + f_\alpha \delta\alpha$, и определив $\psi(t)$ решением краевой задачи (мы ограничимся функционалом первого типа, так как для остальных исключение проводится точно так же)

$$\left(\frac{d}{dt} - f_x[t] \right)^* \psi = \Phi_x[t]; \quad \Gamma_x^* \psi = 0,$$

получим

$$\int_0^T \Phi_x[t]\delta x(t)dt = \int_0^T \psi \frac{\partial f}{\partial u} \delta u dt + \left(\int_0^T \psi \frac{\partial f}{\partial \alpha} dt, \delta\alpha \right) - \psi \delta x|_0^T.$$

Последнее слагаемое $\psi \delta x|_0^T$ преобразуется с помощью соотношений $\Gamma_x^* \psi = 0$, $\Gamma_x \delta x + \Gamma_\alpha \delta\alpha = 0$ в выражение $\psi \delta x|_0^T = (\bar{a}, \delta\alpha)$, где $\bar{a} = \{\bar{a}^1, \bar{a}^2, \dots, \bar{a}^p\}$, компоненты вектора \bar{a} — суть некоторые линейные функционалы от $\psi(\cdot)$, обычно легко вычисляемые, коль скоро

$\psi(t)$ найдено решением соответствующей краевой задачи. Проверку этого факта мы пока отложим с тем, чтобы скорее дойти до нужного результата. Итак, получаем выражение

$$\begin{aligned} \delta F[\delta u(\cdot), \delta \alpha] = & \int_0^T \Phi_u[t] \delta u(t) dt + \int_0^T f_u^*[t] \psi(t) \delta u(t) dt + \\ & + \left(\int_0^T f_\alpha^*[t] \psi(t) dt, \delta \alpha \right) + \left(\int_0^T \Phi_\alpha[t] dt, \delta \alpha \right) - (\bar{a}, \delta \alpha). \end{aligned}$$

Обозначив

$$\frac{\partial F[u(\cdot), \alpha]}{\partial u(\cdot)} = w(t) = \Phi_u[t] + f_u^*[t] \psi(t),$$

$$\frac{\partial F[u(\cdot), \alpha]}{\partial \alpha} = a = \int_0^T f_\alpha^*[t] \psi(t) dt - \bar{a},$$

получим выражение для вариации функционала через вариации $\{\delta u(\cdot), \delta \alpha\}$:

$$\delta F[\delta u(\cdot), \delta \alpha] = \int_0^T w(t) \delta u(t) dt + (a, \delta \alpha). \quad (5)$$

4. Конус вариаций независимого аргумента теперь есть «произведение» обычного конуса K_u и конуса K_α возможных вариаций $\delta \alpha$; если на значения параметров α никаких условий не наложено, K_α есть все p -мерное линейное пространство. Именно этот случай мы и будем иметь в виду.

5. Конус смещений K_F строится обычным образом: вычислив производные всех входящих в постановку задачи функционалов $F_0, F_1, \dots, F_m[u(\cdot)]$, получаем для вектора смещения $\delta F = \{\delta F_0, \delta F_1, \dots, \delta F_m\}$ формулу

$$\delta F[\delta u(\cdot), \delta \alpha] = \int_0^T W(t) \delta u(t) dt + A \delta \alpha, \quad (6)$$

и K_F есть задаваемое правой частью (6) отображение конуса вариаций $K_u \times K_\alpha$ в E_{m+1} . Его замыкание — выпуклый конус.

6. Если траектория оптимальна, то существует определяемая вектором $g = \{-1, g^1, \dots, g^m\}$ гиперплоскость, опорная к конусу K_F в его вершине, т. е.

$$(\delta F, g) \leq 0 \quad \text{для всех} \quad \delta F \in K_F.$$

Учитывая (6), получаем

$$\left(g, \int_0^T W \delta u dt\right) + (g, A \delta \alpha) = \int_0^T (W^* g, \delta u) dt + (A^* g, \delta \alpha) \leq 0$$

для всех $\delta u(\cdot) \in K_u$, $\delta \alpha \in K_\alpha$. Это условие в силу независимости $\delta u(\cdot)$ и $\delta \alpha$ превращается в два: условие

$$(w^*(t)g, \delta u) \leq 0 \quad \text{для всех } \delta u \in K_t, \quad t \in [0, T] \quad (7)$$

приводит к обычной формулировке принципа максимума (как в § 5 и § 6). Кроме того, получаем *общее условие трансверсальности*

$$A^* g = 0. \quad (8)$$

Условие (8) должно выполняться на оптимальной траектории; оно представляет собой p конечных соотношений. Матрица A ($m+1$ строка, p столбцов) однозначно вычисляется на любой фиксированной исследуемой траектории $\{u(\cdot), \alpha, x(\cdot)\}$; это относится и к матрице влияния $W(t)$.

2. Общие двухточечные краевые условия. Используя формальную запись $\Gamma(x, \alpha) = 0$, естественно ограничиться условиями вида $G[x(0), x(T), \alpha] = 0$, где $G = \{G^1, G^2, \dots, G^n\}$. В этом случае условия $\Gamma_x \delta x + \Gamma_\alpha \delta \alpha = 0$ имеют форму $B \delta x(0) + C \delta x(T) + D \delta \alpha = 0$, где $B = \frac{\partial}{\partial x(0)} G[x(0), x(T), \alpha]$, $C = \frac{\partial}{\partial x(T)} G[x(0), x(T), \alpha]$ — $(n \rightarrow n)$ -матрицы, $D = G_\alpha[x(0), x(T), \alpha]$ — $(p \rightarrow n)$ -матрица; все они однозначно определяются на исследуемой траектории $\{u(\cdot), \alpha, x(\cdot)\}$. Объединяя $\delta x(0)$, $\delta x(T)$ в единый $2n$ -вектор $z = \{\delta x(0), \delta x(T)\}$ и вводя $(2n \rightarrow n)$ -матрицу $P = \{B; C\}$, запишем краевое условие для δx в виде

$$Pz + D\delta\alpha = 0.$$

Выделив n компонент в z так, чтобы $(n \rightarrow n)$ -матрица соответствующих столбцов P имела обратную, обозначив эту часть матрицы P_1 , а остальную — P_2 , запишем условия в виде

$$P_1 z_1 + P_2 z_2 + D\delta\alpha = 0 \quad \text{или} \quad z_1 = -P_1^{-1}(P_2 z_2 + D\delta\alpha). \quad (9)$$

Сопряженные краевые условия для ψ получим, разбив $2n$ -вектор $\varphi = \{\psi, (0), \psi(T)\}$ на две части и выделив в вектор φ_1 те же компоненты, что и в z_1 , а остальные — в φ_2 . Тогда (см. § 3) условия $\Gamma_x^* \psi = 0$ подбираются так, чтобы из $\Gamma_x^* \psi = 0$ и $z_1 = -P_1^{-1}P_2 z_2$ следовало

$$\begin{aligned} (z_1, \varphi_1) + (z_2, \varphi_2) &= (z_2, \varphi_2) - (P_1^{-1}P_2 z_2, \varphi_1) = \\ &= (z_2, \varphi_2 - [P_1^{-1}P_2]^* \varphi_1) = 0 \quad (\text{при любом } z_2). \end{aligned}$$

Однако в задаче с параметрами мы имеем другую, неоднородную, связь z_1 с z_2 (9), учитывая которую получаем

$$(\delta x, \psi)|_0^T = (z_1, \varphi_1) + (z_2, \varphi_2) = (z_2, \varphi_2) - (P_1^{-1}P_2 z_2, \varphi_1) - \\ - (P_1^{-1}D \delta x, \varphi_2) = -(P_1^{-1}D \delta x, \varphi_2) = -([P_1^{-1}P_2]^* \varphi_2, \delta x).$$

Таким образом, вектор \tilde{a} в выражении для δF вычисляется по формуле

$$\tilde{a} = [P_1^{-1}P_2]^* \varphi_2. \quad (10)$$

3. Общие краевые условия. Общие краевые условия для системы $x = f(x, u)$ могут быть записаны в форме

$$F_i[x(\cdot)] = 0, \quad i = 1, 2, \dots, n,$$

где $F_i[x(\cdot)]$ — некоторые функционалы, вычисление которых требует знания лишь фазовой траектории. Особенностью управляемой системы является то, что эти функционалы не должны зависеть от мгновенных значений $\dot{x}(t)$ в каких-то точках t', t'', \dots , т. е. по существу, от значений $u(t)$ на множестве меры нуль. Точный смысл этому ограничению можно придать, потребовав выполнения условия (для любой пары $x(\cdot), y(\cdot)$):

$$|F[x(\cdot)] - F[y(\cdot)]| \leq C \max_t \|x(t) - y(t)\|. \quad (11)$$

В связи с этим стоит заметить, что первичная постановка вариационной задачи может иметь такой вид:

Найти функции $u(\cdot), x(\cdot)$, связанные уравнением $\dot{x} = f(x, u)$, минимизирующие значение функционала $F_0[x(\cdot), u(\cdot)]$ при условиях: $u^*(t) \in U, F_i[x(\cdot), u(\cdot)] = 0, i = 1, 2, \dots, M$.

Выделив n условий $F_i = 0$ и назвав их «краевыми условиями» $\Gamma(x) = 0$, придем к обычной постановке задачи. Разумеется, это выделение не является совсем произвольным: кроме формального требования — число условий должно быть равно размерности x , — следует иметь в виду и содержательное: выделенные условия должны при заданной функции $u(\cdot)$ обеспечить существование и единственность краевой задачи

$$\dot{x} = f(x, u(t)); \quad \Gamma(x) = 0.$$

Есть еще одно важное для приближенного решения вариационной задачи требование: для решения полученной краевой задачи должен существовать эффективный численный алгоритм. Известно, что проще всего в этом отношении задача Коши. Поэтому часто бывает удобно формально ввести параметры α в постановку задачи, записав краевые условия $\Gamma(x, \alpha) = 0$ в виде, например, $x(0) - \alpha = 0$ (если среди условий первичной постановки задачи есть условия, фиксирующие значения некоторых компонент $x(0)$, то соответствующие компоненты α не варьируются). Имея в виду этот прием,

мы не рассматриваем краевых условий более сложных, чем двухточечные. Однако совсем отказаться от общих условий и иметь дело только с задачей Коши не удастся. Дело в том, что в приложениях существуют уравнения, для которых сравнительно просто решается краевая двухточечная задача, тогда как решение задачи Коши невероятно трудно. Именно такой задачей является задача о защите от излучения, выходящего из ядерного реактора. Вариационная задача для защиты решалась автором, и ниже она будет подробно описана. Здесь мы поясним суть дела на простеньком примере, моделирующем основные черты этой задачи,

$$\begin{aligned} \frac{dx^1}{dt} &= a[u(t)]x^2; \quad \frac{dx^2}{dt} = b[u(t)]x^1, \quad 0 \leq t \leq 1, \\ x^1(0) &= X_1; \quad x^2(1) = 0. \end{aligned} \quad (12)$$

Характерным обстоятельством, определяющим выбор численного алгоритма и возникающие вычислительные трудности, является величина коэффициентов a и b ; в задаче о защите $a(u) \simeq b(u) \simeq 30-40$. Если формально ввести условие с параметром $x^2(0) = \alpha$, а $x^2(1) = 0$ отнести к дополнительным условиям задачи типа $F[u(\cdot)] = 0$, ограничивающим возможные функции $u(\cdot)$, то внешне простая и бесхитростная задача Коши практически не поддается численному решению на современных ЭВМ: дело в том, что общее решение этой системы состоит из двух качественно совершенно разных компонент; одна из них — типа сильно растущей экспоненты e^{40t} , вторая — типа сильно убывающей e^{-40t} . Поэтому попытка подбором α «попасть» в правое краевое условие сопряжена с большими вычислительными трудностями: $\frac{\partial x^2(1)}{\partial \alpha} \simeq e^{40} \simeq 10^{16}$. Кроме того, интегрирование задачи Коши сопровождается аналогичным ($\sim e^{40t}$) ростом влияния допущенных у левого конца вычислительных ошибок. В то же время решение (12) как краевой задачи методом прогонки совершенно элементарно и хорошо освоено в современной вычислительной практике (см. [23], [24]).

4. Традиционные условия трансверсальности. Для того чтобы установить связь с общепринятыми условиями трансверсальности, рассмотрим частную задачу: дана система

$$\frac{dx}{dt} = f(x, u), \quad 0 \leq x \leq T, \quad u \in U,$$

с условиями

$$\begin{aligned} G_k[x(0)] &= 0, & k &= 1, 2, \dots, K < n, \\ D_i[x(T)] &= 0, & i &= 1, 2, \dots, m < n; \end{aligned}$$

минимизировать $D_0[x(T)]$.

Введем условия $\Gamma(x, \alpha) = 0$ в виде $x(0) - \alpha = 0$ и проварируем некоторую траекторию $\{u(\cdot), \alpha, x(\cdot)\}$, для которой выполнены все условия $G_k = D_i = 0$. Воспользуемся результатами анализа общей задачи с параметрами, для чего следует прежде всего решить краевую задачу ($m+1$ раз):

$$\frac{d\psi^{(i)}}{dt} + f_x^*[t]\psi^{(i)} = 0; \quad \psi^{(i)}(T) = \frac{\partial D_i}{\partial x} \Big|_{x(T)}, \quad i = 0, 1, \dots, m;$$

это позволяет вычислить вариацию соответствующего функционала $F_i[u(\cdot), \alpha] \equiv D_i[x(T)]$:

$$\delta F_i[\delta u(\cdot), \delta \alpha] = \int_0^T \psi^{(i)} f_u[t] \delta u dt + (\psi^{(i)}(0), \delta \alpha).$$

Образует функцию $\psi(t) = \sum_{i=0}^m \psi^{(i)}(t) g^i$, являющуюся решением задачи

$$\frac{d\psi}{dt} + f_x[t]\psi = 0; \quad \psi(T) = \sum_{i=0}^m g^i \frac{\partial D_i[x(T)]}{\partial x}.$$

Матрица A в формуле (6) — это матрица, строки которой образованы n -векторами $\psi^{(i)}(0)$; таким образом,

$$A^*g = \sum_{i=0}^m g^i \psi^{(i)}(0) = \psi(0),$$

и условие трансверсальности будет получено из условия $0 = (A^*g, \delta \alpha) = (\psi(0), \delta \alpha)$, в котором, однако, надо учесть, что компоненты $\delta \alpha$ не являются независимыми: они связаны K условиями $G_k(\alpha + \delta \alpha) = 0$, т. е. $\frac{\partial G_k}{\partial \alpha} \delta \alpha = 0$, $k = 1, 2, \dots, K$, или, в компактной форме, $B \delta \alpha = 0$, где B — матрица $n \rightarrow K$, строки которой есть градиенты функций G_k . Предполагая условия $G_k[x(0)] = 0$ невырожденными, мы можем выделить из B K столбцов, образующих невырожденную матрицу; выделив соответствующие компоненты $\delta \alpha$ в K -вектор $\delta \alpha_1$, а остальные — в $(n - K)$ -вектор $\delta \alpha_2$, запишем связь $B \delta \alpha = 0$ в виде $B_1 \delta \alpha_1 + B_2 \delta \alpha_2 = 0$ или $\delta \alpha_1 = -B_1^{-1} B_2 \delta \alpha_2$.

Выделив соответствующие компоненты $\psi(0)$ в векторы φ_1 и φ_2 , условие трансверсальности запишем в виде

$$0 = (\psi(0), \delta \alpha) = (\varphi_1, \delta \alpha_1) + (\varphi_2, \delta \alpha_2) = (\varphi_1, -A_1^{-1} A_2 \delta \alpha_2) + (\varphi_2, \delta \alpha_2) = (\delta \alpha_2, \varphi_2 - [B_1^{-1} B_2]^* \varphi_1).$$

Так как $\delta \alpha_2$ уже оказывается вектором с независимыми компонентами, то условие трансверсальности получает форму краевого условия для $\psi(0)$:

$$\varphi_2 = -[B_1^{-1} B_2]^* \varphi_1.$$

Это есть $n-K$ соотношений между компонентами $\psi(0)$. Впрочем, чаще такие условия формулируются иначе: условиями $G_k[x(0)] = 0$ в n -мерном пространстве выделяется $(n-K)$ -мерное гладкое многообразие; $\delta\alpha = \delta x(0)$ и подлежат рассмотрению не всевозможные n -мерные векторы $\delta\alpha$, а лишь те, которые лежат в касательной к упомянутому многообразию $(n-K)$ -мерной гиперплоскости (касательной в точке $x(0)$, разумеется). Тогда условие трансверсальности $(\delta\alpha, \psi(0)) = 0$ означает, что $\psi(0)$ должен быть ортогонален этой касательной $(n-K)$ -мерной гиперплоскости. Это и есть традиционная формулировка условий трансверсальности на левом конце траектории.

5. Задачи со свободным временем T . Пусть длина интервала управления $[0, T]$ — не фиксирована, и T есть «ресурс управления» наряду с функцией $u(\cdot)$. В этом случае удобно (особенно с точки зрения организации расчетов) сделать замену переменных $\tau = t/T$, после чего имеем задачу (1) на фиксированном интервале времени $0 \leq \tau \leq 1$ с параметром T для системы

$$\frac{dx}{d\tau} = Tf(x, u), \quad 0 \leq \tau \leq 1.$$

Все остальное входит в общую схему задачи с параметрами и не заслуживает особого рассмотрения. Заметим, что часто вариацию δT исключают, используя для этого одно из дополнительных условий, например, $F_1[u(\cdot), T]$. Обычным способом определяется вариация

$$\delta F_1[\delta u(\cdot), \delta T] = \int_0^1 w_1(\tau) \delta u(\tau) d\tau + b_1 \delta T.$$

Если $b_1 \neq 0$, то $\delta T = -\frac{1}{b_1} \int_0^1 w_1(t) \delta u(t) dt$, и для всех остальных функционалов $F_i[u(\cdot)]$ вариация может быть написана только в терминах $\delta u(\cdot)$:

$$\delta F[\delta u(\cdot), \delta T] = \int_0^1 w(\tau) \delta u(\tau) d\tau + b \delta T = \int_0^1 \left[w(\tau) - \frac{b}{b_1} w_1(\tau) \right] d\tau.$$

6. Склерономные системы. Пусть t входит явно в правую часть системы уравнений; пусть также область U зависит от времени, таким образом,

$$\frac{dx}{dt} = f(x, u, t), \quad u(t) \in U(t).$$

С этим обобщением никаких осложнений не связано, если зависимость $U(t)$ — непрерывная. Зависимость $f(x, u, t)$ от t может быть, в сущности, произвольной.

7. Задачи с разрывной правой частью [18]. Пусть уравнение движения управляемой системы имеет вид

$$\frac{dx}{dt} = \begin{cases} f_1(x, u) & \text{при } R[x(t)] < 0, \\ f_2(x, u) & \text{при } R[x(t)] > 0, \end{cases} \quad (13)$$

где $R(x)=0$ — уравнение гладкой поверхности разрыва. Ради простоты мы ограничимся лишь одной поверхностью разрыва. Кроме того, предположим, что в точке пересечения траектории с поверхностью $R(x)=0$ выполнены условия: при всех u $R_x f_1(x, u) > 0$, $R_x f_2(x, u) > 0$. Этим запрещаются так называемые «скользящие» режимы движения $x(t)$ вдоль поверхности разрыва и обеспечивается применимость теории возмущений: малое возмущение траектории приводит к соответственно малому же изменению момента пересечения траекторией $x(t)$ поверхности $R(x)=0$.

При решении подобных задач удобно (особенно с точки зрения организации численных алгоритмов) произвести замену времени так, чтобы разрыв правой части происходил в фиксированный в новом времени момент. Так от системы $\dot{x}=f(x, u)$ переходим к системе

$$\frac{dx}{dt} = f(x, u, t), \quad 0 \leq t \leq 2, \quad (13^*)$$

где

$$f(x, u, t) = \begin{cases} \alpha_1 f_1(x, u) & \text{при } 0 \leq t \leq 1, \\ \alpha_2 f_2(x, u) & \text{при } 1 < t \leq 2, \end{cases}$$

однако к дополнительным условиям задачи добавляется еще одно:

$$F_{m+1}[u(\cdot), \alpha_1, \alpha_2] \equiv R[x(1)] = 0.$$

Задача с разрывной правой частью, таким образом, сведена к стандартной — зависимость правой части от t , как уже отмечалось, никаких осложнений не вносит. На этом можно было бы и закончить анализ, однако некоторые методы приближенного решения вариационных задач «болезненно» реагируют на увеличение числа дополнительных условий; поэтому мы проведем выкладки, позволяющие избежать этого. И в этом случае мы ограничимся вычислением производной определенного на решении системы (13, 13*) функционала

$$F[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt,$$

точнее, вычислением вектор-функции $w(t)$ в формуле

$$\int_0^T Y[t] \delta x(t) dt = \int_0^T w(t) \delta u(t) dt.$$

Пусть траектория $x(t)$ пересекает поверхность разрыва в момент t^* : $R[x(t^*)] = 0$. Уравнение в вариациях для $\delta x(t)$ имеет вид

$$\frac{d\delta x}{dt} - \frac{\partial f_1}{\partial x} \delta x = \frac{\partial f_1}{\partial u} \delta u \quad \text{на } [0, t^*),$$

$$\frac{d\delta x}{dt} - \frac{\partial f_2}{\partial x} \delta x = \frac{\partial f_2}{\partial u} \delta u \quad \text{на } (t^*, 2].$$

Пусть возмущенная траектория $x(t) + \delta x(t)$ пересекает $R = 0$ в момент $t^* + \delta$ (пока для определенности будем считать $\delta \geq 0$).

Тождество Лагранжа запишем в виде

$$\begin{aligned} & \int_0^{t^*} \left[\psi \left(\frac{d\delta x}{dt} - \frac{\partial f_1}{\partial x} \delta x \right) + \delta x \left(\frac{d\psi}{dt} + \frac{\partial f_1^*}{\partial x} \psi \right) \right] dt + \\ & + \int_{t^*+\delta}^T \left[\psi \left(\frac{d\delta x}{dt} - \frac{\partial f_2}{\partial x} \delta x \right) + \delta x \left(\frac{d\psi}{dt} + \frac{\partial f_2^*}{\partial x} \psi \right) \right] dt = \psi \delta x \Big|_0^T - \psi \delta x \Big|_{t^*+\delta}^{t^*+\delta}. \end{aligned}$$

Мы предположим, что $Y[t]$ ограничена в точке t^* , поэтому

$$\int_{t^*}^{t^*+\delta} Y[t] \delta x(t) dt = O(\|\delta u\|^2), \quad \text{так как } \delta \simeq O(\|\delta u\|).$$

Для того чтобы осуществить стандартное исключение вариации δx , нужно в тождестве Лагранжа ликвидировать «лишнее» слагаемое $\psi \delta x|_{t^*+\delta}^{t^*+\delta}$. Это приведет к разрыву функции $\psi(t)$ в точке t^* с определенными соотношениями на разрыве. Обозначим через $\{v(\cdot), y(\cdot)\}$ возмущенную траекторию. Тогда, с точностью до $O(\|\delta u\|^2)$,

$$\begin{aligned} y(t^* + \delta) &= y(t^*) + \delta f_1(y^*, v^*), \\ x(t^* + \delta) &= x(t^*) + \delta f_2(x^*, u^*). \end{aligned}$$

Вычитая, получим связь между $\delta x(t^*)$ и $\delta x(t^* + \delta)$:

$$\delta x(t^* + \delta) = \delta x(t^*) + \delta [f_1(y^*, v^*) - f_2(x^*, u^*)].$$

(Здесь обозначено: $y^* = y(t^*)$, $x^* = x(t^*)$, $u^* = u(t^* + 0)$, $v^* = v(t^* - 0) = u(t^* - 0) + O(\|\delta u\|)$.)

Следует еще использовать связь между δ и $\delta x(t^*)$:

$$\begin{aligned} R[y(t^* + \delta)] &= 0 = R[y^* + \delta f_1(y^*, v^*)] = \\ &= R[x^* + \delta x^* + \delta f_1(y^*, v^*)] = R(x^*) + R_x \delta x^* + \delta R_x f_1(y^*, v^*), \end{aligned}$$

и так как $R(x^*) = 0$, то

$$\delta = -\frac{R_x \delta x^*}{R_x f_1(y^*, v^*)} = -\frac{R_x \delta x^*}{R_x f_1(x^*, v^*)}.$$

Теперь соотношение между $\delta x(t^*)$ и $\delta x(t^* + \delta)$ примет вид

$$\delta x(t^* + \delta) = \delta x(t^*) - \frac{R_x \delta x^*}{R_x f_1} [f_1 - f_2],$$

где $f_1 = f_1[x^*, u(t^* - 0)]$, $f_2 = f_2[x^*, u(t^* + 0)]$. Обращение в нуль (с точностью до $O(\|\delta u\|^2)$), что, собственно, и нужно для вычисления производной функционала) выражения

$$\begin{aligned} \psi \delta x|_{t^*+\delta} &= \psi(t^* + \delta) \delta x(t^* + \delta) - \psi(t^*) \delta x(t^*) = \\ &= (\psi^+ - \psi^-, \delta x^*) - \frac{R_x \delta x^*}{R_x f_1} (f_1 - f_2, \psi^+) = \left(\psi^+ - \psi^- - \frac{(f_1 - f_2, \psi^+)}{R_x f_1} R_x, \delta x^* \right) \end{aligned}$$

будет обеспечено для любых δx^* , если $\psi^+ = \psi(t^* + 0)$ и $\psi^- = \psi(t^* - 0)$ удовлетворяют условию скачка:

$$\psi^- = \psi^+ - \frac{(f_1 - f_2, \psi^+)}{R_x f_1} R_x. \quad (14)$$

Таким образом, вычисление производной функционала осуществляется стандартно, только функция $\psi(t)$ определяется как решение уравнений

$$\begin{aligned} \frac{d\psi}{dt} + \frac{\partial f_2^*}{\partial x} \psi &= -Y[t] \quad \text{на } (t^*, T), \\ \frac{d\psi}{dt} + \frac{\partial f_1^*}{\partial x} \psi &= -Y[t] \quad \text{на } (0, t^*), \\ \Gamma_x^* \psi &= 0, \end{aligned}$$

с условием скачка (14) при $t = t^*$.

Если условия $\Gamma(x) = 0$ — суть данные Коши $x(0) - X_0 = 0$ (а это наиболее часто встречающийся в приложениях случай), то $\Gamma_x^* \psi = 0$ — данные Коши для $\psi(T)$, и численное интегрирование краевой задачи для $\psi(t)$ не встречает никаких затруднений. Форма записи условий скачка (14) в этом случае особенно удобна, так как определяет в явном виде переход от $\psi(t^* + 0)$ к $\psi(t^* - 0)$ при интегрировании справа налево.

Мы рассмотрели выше лишь случай $\delta \geq 0$. Точно таким же образом проводится и анализ при $\delta \leq 0$; очевидным образом меняя местами правые и левые значения, получим соотношение

$$\psi^+ = \psi^- - \frac{(f_2 - f_1, \psi^-)}{R_x f_2} R_x. \quad (14^*)$$

Предоставим читателю самому убедиться в эквивалентности соотношений (14) и (14*). В случае большего числа поверхностей разрыва никаких новых обстоятельств не возникает.

8. **Дополнительные условия типа неравенства.** Пусть дополнительные условия в терминах дифференцируемых функционалов имеют вид:

$$\begin{aligned} F_i[u(\cdot)] &\leq 0, & i = 1, 2, \dots, m_1 & \quad (m_1 \leq m); \\ F_i[u(\cdot)] &= 0, & i = m_1 + 1, \dots, m. \end{aligned}$$

В этом случае весь анализ проводится стандартно, однако, необходимым условием оптимальности исследуемой траектории является отсутствие в выпуклом конусе K_F смещений δF элементов так называемого конуса запрещенных смещений K_s . Этот конус в $(m+1)$ -мерном пространстве описывается следующим образом:

$$\begin{aligned} 1) & \quad \delta F_0 < 0, \\ 2) & \quad \delta F_i \leq 0, \quad i = 1, 2, \dots, m_1, \\ 3) & \quad \delta F_i = 0, \quad i = m_1 + 1, \dots, m. \end{aligned}$$

Содержательный смысл K_s очевиден: если вариация управления $\delta u(\cdot)$ порождает смещение $\delta F \in K_s$, то управление не оптимально, поскольку в первом порядке при переходе от $u(\cdot)$ к $u(\cdot) + \delta u(\cdot)$ δF_0 уменьшается, а дополнительные условия не нарушаются. В рассмотренной в § 5 задаче K_s состоял из одного луча $e = \{-1, 0, \dots, 0\}$. Выпуклость K_s — очевидна. Если выпуклые конусы K_F и K_s не пересекаются, то они могут быть разделены гиперплоскостью с нормалью g :

$$\begin{aligned} (g, \delta F) &\leq 0 & \text{для } \delta F \in K_F, \\ (g, \delta F) &\geq 0 & \text{для } \delta F \in K_s. \end{aligned}$$

Первое соотношение приводит к обычной формулировке принципа максимума, второе дает дополнительную информацию о компонентах вектора g . В самом деле, $e = \{-1, 0, \dots, 0\} \in K_s$, следовательно, $(g, e) = -g^0 > 0$, т. е. $g^0 < 0$, или, нормируя g , $g^0 = -1$. Далее, $e^{(i)} = \{0, \dots, 0, -1_i, 0, \dots, 0\} \in K_s$ (1_i означает 1 на i -м месте, $i = 1, 2, \dots, m_1$). Следовательно,

$$(e^{(i)}, g) = -g^i \geq 0, \quad \text{т. е. } g^i \leq 0 \quad \text{при } i = 1, 2, \dots, m_1.$$

Этим исчерпывается информация, которую можно извлечь из условия $(g, \delta F) \geq 0$ для $\delta F \in K_s$, так как K_s есть выпуклая оболочка векторов $e, e^{(1)}, e^{(2)}, \dots, e^{(m_1)}$.

9. **Задачи для уравнений с запаздыванием [24].** Рассмотрим вариационную задачу, в которой управление определяет фазовую траекторию системы задач Коши для уравнения с запаздыванием:

$$\frac{dx}{dt} = f[\tilde{F}[x(\cdot), t], x(t), u(t)]; \quad 0 \leq t \leq T. \quad (15)$$

Здесь $\tilde{F}[x(\cdot), t]$ — некоторый заданный функционал, зависящий от значений функции $x(\cdot)$ лишь на интервале $[t-h, t]$, $h > 0$ —

заданное число. Этот функционал предположим дифференцируемым; пусть определяющее \bar{F} конкретное выражение прямым варьированием дает формулу

$$\delta \bar{F} [\delta x(\cdot), t] = \int_{t-h}^t Z(t, \tau) \delta x(\tau) d\tau.$$

Функция $Z(t, \tau)$ вычисляется, разумеется, на исследуемой траектории $\{u(\cdot), x(\cdot)\}$ и может содержать особенности типа δ -функции. Областью определения функции $Z(t, \tau)$ является полоса

$$[0 \leq t \leq T] \times [t-h < \tau \leq t].$$

В качестве данных Коши задается условие *)

$$x(t) = \varphi(t), \quad -h \leq t < 0, \quad x(0) = \varphi(0).$$

Функция $\varphi(t)$ либо фиксирована, либо является искомым элементом управления; мы будем иметь в виду этот последний, более общий случай. Если $\varphi(t)$ — фиксированная функция, $\delta\varphi(t)$ следует считать нулем. Пусть $F[u(\cdot), \varphi(\cdot), \varphi(0)]$ — некоторый дифференцируемый функционал, определенный на траектории системы (15); нашей целью является вычисление его производной. Основным моментом этого вычисления является вывод формулы типа

$$\int_0^T Y[t] \delta x(t) dt = \int_0^T w(t) \delta u(t) dt + \int_{-h}^0 a(t) \delta \varphi(t) dt + (a, \delta \varphi(0)), \quad (16)$$

где $Y[t]$ — известная функция, определенная на исследуемой невозмущенной траектории $\{\varphi(\cdot), \varphi(0), u(\cdot), x(\cdot)\}$, а $w(t)$, $a(t)$, a подлежат определению. Основные элементы техники вычислений стандартны.

1. Уравнение в вариациях, определяющее $\delta x(t)$:

$$\begin{aligned} \frac{d\delta x}{dt} &= f_F[t] \delta \bar{F} + f_x[t] \delta x + f_u[t] \delta u = \\ &= f_F[t] \int_{t-h}^t Z(t, \tau) \delta x(\tau) d\tau + f_x[t] \delta x + f_u[t] \delta u; \\ \delta x(t) &= \delta \varphi(t), \quad -h \leq t < 0; \quad \delta x(0) = \delta \varphi(0). \end{aligned}$$

2. Тождество Лагранжа:

$$\int_0^T \left[\psi \left(\frac{d\delta x}{dt} + f_x \delta x \right) + \delta x \left(\frac{d\psi}{dt} - f_x^* \psi \right) \right] dt = \psi \delta x \Big|_0^T.$$

*) Известно, что $\varphi(0)$ играет особую роль среди всех значений $\varphi(t)$; поэтому его естественно выделить как отдельный объект.

Используя уравнение в вариациях, получим

$$\int_0^T \psi (\delta \dot{x} - f_x \delta x) dt = \int_0^T \psi(t) f_F[t] \int_{t-h}^t Z(t, \tau) \delta x(\tau) d\tau dt + \int_0^T \psi f_u \delta u dt.$$

Первое слагаемое правой части подлежит дальнейшим преобразованиям (рис. 6):

$$\int_0^T \psi(t) f_F[t] \int_{t-h}^t Z(t, \tau) \delta x(\tau) d\tau dt = \iint_{OABC} \psi(t) f_F[t] Z(t, \tau) \delta x(\tau) dt d\tau.$$

Будем считать $\psi(t)$ определенной на интервале $0 \leq t \leq T+h$, причем $\psi(t) \equiv 0$ при $t \in [T, T+h]$. Тогда $\iint_{OABC} = \iint_{ODC} + \iint_{OABD}$ (так как

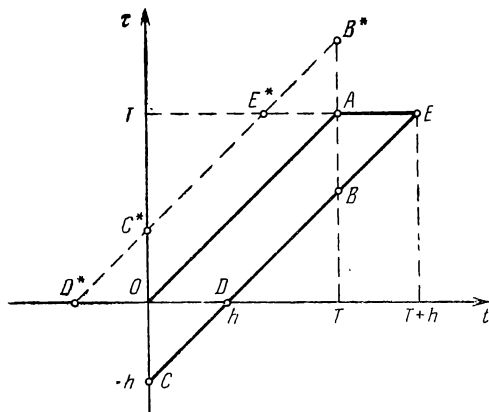


Рис. 6.

в AEB $\psi \equiv 0$, то остальные функции — f_F , Z , δx — можно считать доопределенными в этой области любым образом). Итак,

$$\begin{aligned} \int_0^T dt \psi(t) f_F[t] \int_{t-h}^t Z(t, \tau) \delta x(\tau) d\tau &= \iint_{OABC} = \\ &= \iint_{ODC} + \int_0^T d\tau \int_{\tau}^{\tau+h} \psi(t) f_F[t] Z(t, \tau) dt \delta x(\tau) = \\ &= \int_0^h dt \psi(t) f_F[t] \int_{t-h}^0 Z(t, \tau) \delta \varphi(\tau) d\tau + \int_0^T dt \delta x(t) \int_t^{t+h} \psi(\tau) f_F[\tau] Z(\tau, t) d\tau. \end{aligned}$$

Проводя эти выкладки, следует иметь в виду, что δx , Z , f_F , ψ , $\delta \varphi$ — век-

торы той же размерности, что и x ; $Z\delta x$ — скаляр, ψf_F — тоже. Легко видеть, что осуществленные в преобразованиях перестановки функций законны.

Из тождества Лагранжа получаем соотношение

$$\int_0^T \left\{ \psi(t) f_u[t] \delta u(t) + \delta x(t) \left[\frac{d\varphi}{dt} + f_x^* \psi + \int_t^{t+h} \psi(\tau) f_F(\tau) Z(\tau, t) d\tau \right] \right\} dt + \\ + \int_{-h}^0 \delta \varphi(\tau) \int_0^{h+\tau} (\psi(t), f_F(t)) Z(t, \tau) dt d\tau + \psi(0) \delta \varphi(0) = \psi(T) \delta x(T).$$

Теперь видно, что цель будет достигнута, если в качестве $\psi(t)$ взять решение задачи Коши для уравнения

$$\frac{d\psi}{dt} + f_x^*[t] \psi + \int_t^{t+h} (\psi(\tau), f_F[\tau]) Z(\tau, t) d\tau = -Y[t], \quad (17) \\ 0 \leq t \leq T,$$

с начальными данными

$$\psi(t) = 0 \quad \text{при} \quad t \in (T, -T + h], \quad \psi(T) = 0.$$

Если вариация функционала включает слагаемое вида $(b, \delta x(T))$, то для граничного значения ψ естественно взять $\psi(T) = b$. Уравнение для $\psi(t)$, решаемое справа налево, есть обычное уравнение с запаздыванием. Итак, формула (16) получена, причем

$$w(t) \equiv f_u^*[t] \psi(t), \\ a(t) \equiv \int_0^{h+t} (\psi(\tau), f_F[\tau]) Z(\tau, t) d\tau, \quad -h \leq t \leq 0, \\ a = \psi(0).$$

Стоит заметить, что $Z(\tau, t)$ определена в полосе $[0 \leq \tau \leq T] \times [\tau - h \leq t \leq \tau]$, т. е. в OAE^*D^* . Для того чтобы уравнение (17) было полностью определено, нужно знать $Z(\tau, t)$ в области $[0 \leq t \leq T] \times [t \leq \tau \leq t + h]$, т. е. в OC^*B^*A ; как уже было отмечено, в E^*B^*A , т. е. при $\tau > T$, можно доопределить $Z(\tau, t)$, например, нулем. Вычисление $a(t)$ осуществляется интегрированием $Z(\tau, t)$ в $[-h \leq t \leq 0] \times [0 \leq \tau \leq h + t]$, т. е. в OC^*D^* , где Z определена.

§ 8. Принцип максимума в задачах с фазовыми ограничениями

В § 5 при выводе принципа максимума мы ограничились задачами с дифференцируемыми по Фреше функционалами $F_i[u(\cdot)]$. Здесь для простоты мы рассмотрим задачу с одним функционалом, дифференцируемым лишь по направлениям в функциональном

пространстве. Итак, ищется управление $u(\cdot)$ в задаче:

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (1)$$

на решениях системы

$$\dot{x} = f(x, u), \quad \Gamma(x) = 0, \quad u \in U, \quad 0 \leq t \leq T, \quad (2)$$

при условиях

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m, \quad (3)$$

$$G[x(t)] \leq 0 \quad \text{при всех } t \in [0, T]. \quad (4)$$

Все функционалы $F_0, \dots, F_m[u(\cdot)]$ предположим дифференцируемыми по Фреше, функционал

$$F_{m+1}[u(\cdot)] \equiv \max_t G[x(t)]$$

для наглядности изложения выделен из стандартной системы обозначений. $G(\xi)$ — скалярная непрерывно дифференцируемая функция n -мерного аргумента ξ . Чтобы разобраться в этой задаче, заменим ее близкой по смыслу, сформулированной в терминах лишь дифференцируемых функционалов. Именно, введем на $[0, T]$ равномерную, например, сетку точек $t_0, t_1, t_2, \dots, t_N$ с шагом $\tau = T/N$, и вместо условия $G[x(t)] \leq 0$ поставим $N+1$ условий

$$G[x(t_j)] \leq 0, \quad j = 0, 1, 2, \dots, N. \quad (4^*)$$

Почти очевидно следующее утверждение:

Л е м м а. *Решение задачи (1), (2), (3), (4*) при достаточно большом N сколь угодно точно аппроксимирует решение исходной задачи в том смысле, что из (4*) следует*

$$G[x(t)] \leq CT/2N \quad \text{при всех } t \in [0, T]. \quad (5)$$

В самом деле, на любой траектории управляемой системы

$$\left| \frac{dG[x(t)]}{dt} \right| = |G_x[x(t)] f(x, u)| \leq C,$$

где C — некоторая постоянная, зависящая от гладкости функции G и от $\|f\|$. Но кривая $G[x(t)]$, имеющая ограниченную производную и неположительная на сетке с шагом $\tau = T/N$, не может стать больше $C\tau/2$. К аппроксимирующей задаче может быть применен анализ § 5, дающий следующий принцип максимума:

Пусть $\psi^{(i)}(t)$, $i = 0, 1, \dots, m$, — решения уравнений

$$\frac{d\psi^{(i)}}{dt} + f_x^*[t] \psi^{(i)} = -Y^{(i)}[t]; \quad \Gamma_x^* \psi^{(i)} = 0,$$

где $Y^{(i)}[t]$ — функции, появляющиеся при прямом варьировании

определяющих функционалы F_i выражений:

$$\delta F_i [\delta u(\cdot)] = \int_0^T \tilde{w}^{(i)}(t) \delta u(t) dt + \int_0^T Y^{(i)}[t] \delta x(t) dt.$$

Пусть $\psi(t, t_j)$ — решения уравнений

$$\frac{d\psi(t, t_j)}{dt} + f_x^*[t] \psi(t, t_j) = -G_x[x(t_j)] \delta(t - t_j), \quad \Gamma_x^* \psi(\cdot, t_j) = 0.$$

Если траектория $\{u(\cdot), x(\cdot)\}$ аппроксимирующей задачи является оптимальной, то существует вектор $g = \{-1, g^1, \dots, g^m, g(t_1), g(t_2), \dots, g(t_N)\}$ такой, что

$$H[x(t), \psi(t), u(t)] = \max_{u \in U} H[x(t), \psi(t), u], \quad (6)$$

где функция $H[x(t), \psi(t), u]$ строится известным образом:

$$\begin{aligned} H[x(t), \psi(t), u] &\equiv \sum_{i=0}^m g^i \Phi^{(i)}[x(t), u] + \left(\sum_{i=0}^m g^i \psi^{(i)}(t), f[x(t), u] \right) + \\ &+ \left(\sum_{j=0}^N g(t_j) \psi(t, t_j), f[x(t), u] \right) = \sum_{i=0}^m g^i \Phi^{(i)}[x(t), u] + (\psi(t), f[x(t), u]). \end{aligned} \quad (7)$$

Что касается функции $\psi(t)$, то она есть решение определенной с точностью до $N + m$ параметров краевой задачи:

$$\frac{d\psi}{dt} + f_x^*[t] \psi = - \sum_{i=0}^m g^i Y^{(i)}[t] - \sum_{j=1}^N g(t_j) G_x[t_j] \delta(t - t_j), \quad \Gamma_x^* \psi = 0. \quad (8)$$

О величинах $g(t_j)$ известно, кроме того, что $g(t_j) = 0$, если $G[x(t_j)] < 0$ и $g(t_j) \leq 0$, если $G[x(t_j)] = 0$. Можно построить последовательность аппроксимирующих задач с $N_1 < N_2 < \dots \rightarrow \infty$ и перейти к пределу; как нетрудно заметить, это в основном касается уравнения (8), которое будет удобно переписать в виде

$$\frac{d\psi}{dt} + f_x^*[t] \psi = - \sum_{i=0}^m g^i Y^{(i)}[t] + G_x[t] \frac{d}{dt} \sigma_N(t), \quad (8^*)$$

где $\sigma_N(t)$ — монотонно растущая функция, точки роста которой суть лишь те t_j , где $G[x(t_j)] = 0$. Мы не будем подробно рассматривать предельный переход $N \rightarrow \infty$ в системе (8*), а сразу же сформулируем результат:

П р и н ц и п м а к с и м у м а. Если траектория $\{u(\cdot), x(\cdot)\}$ является решением вариационной задачи (1)–(4), то существуют вектор $g = \{-1, g^1, \dots, g^m\}$ и монотонно растущая функция $\sigma(t)$, точки роста которой выделяются условием $G[x(t)] = 0$, такие, что

образованная по формуле (7) функция $H[x(t), \phi(t), u]$, где $\phi(t)$ — решение краевой задачи (8*), при почти всех t достигает максимального в области U значения в точке $u(t)$.

З а м е ч а н и е. При конструировании численных методов решения задач с фазовыми ограничениями (4) мы используем аппроксимацию типа (4*); некоторые дополнительные соображения и следующие из них вычислительные приемы позволяют использовать сравнительно небольшие $N \sim 3-4$, достигая при этом хорошей точности выполнения условия $G[x(t)] \leq 0$ и в тех случаях, когда множество точек $G(x) \approx 0$ занимает значительную часть $[0, T]$. Не случайно выше был рассмотрен лишь функционал (4), и специально отмечалось, что близкий по форме функционал

$$F[u(\cdot)] \equiv \max_t \Phi[x(t), u(t)], \quad (9)$$

который появляется в задачах с ограничениями общего вида $\{G[x(t), u(t)] \leq 0 \text{ при всех } t\}$, требует особого рассмотрения как с теоретической точки зрения, так и при разработке методов приближенного решения. Причиной этого является отсутствие каких бы то ни было свойств гладкости у функции $u(t)$ и, следовательно, у функции $G[x(t), u(t)]$. Поэтому аппроксимация таких условий дискретным набором ограничений

$$G[x(t_j), u(t_j)] \leq 0, \quad j = 1, 2, \dots \quad (10)$$

неэффективна не только при конечном их числе, но и при счетном, всюду плотном на $[0, T]$ множестве точек аппроксимации t_j . Условия (10) могут быть выполнены за счет изменения u на множестве меры нуль, что никак не повлияет на траекторию. Не случайно в п. 2 § 2 использовано более корректное определение

$$F[u(\cdot)] \equiv \operatorname{vrai} \max_t \Phi[x(t), u(t)].$$

Все эти рассуждения на первый взгляд не имеют отношения к приближенному решению задач оптимального управления. Ведь в любой реализации приближенного метода имеют дело не с измеримой функцией, а, например, с кусочно постоянной сеточной. В этом случае разница между функционалами (4) и (9) пропадает, и появляется формальная возможность и для учета (9) использовать аппроксимацию (10). Именно такая точка зрения принята в [31], [68], [75] и других работах, связанных с применением методов математического программирования (см. также §§ 13, 25, 36). К сожалению, этот единообразный подход к объектам разной функциональной природы оплачивается существенным ростом объема вычислений и, вследствие этого, ненадежностью результатов. В данном случае он приведет к очень большому числу точек аппроксимации t_j в (10).

§ 9. Принцип максимума — достаточное условие стационарности траектории

В этом параграфе будет показано, что принцип максимума содержит полную совокупность необходимых условий экстремума первого порядка в том же смысле, в каком для функции двух переменных $f(x, y)$ соотношения $f_x=0$, $f_y=0$ образуют полную систему необходимых условий, а равенство $f_x=0$ является необходимым условием, но полной системы не образует. Более точно этот факт может быть сформулирован следующим образом:

Т е о р е м а. Если траектория управляемой системы $\{u(\cdot), x(\cdot)\}$ удовлетворяет принципу максимума, то она является стационарной.

Аналитический смысл предположения и утверждения этой теоремы будет уточнен в процессе доказательства, которое мы приведем для рассмотренной в § 8 задачи с одним только функционалом, дифференцируемым по Гато, но не по Фреше: $\min_{u(\cdot)} F_0[u(\cdot)]$ для системы $\dot{x}=f(x, u)$, $\Gamma(x)=0$, $u \in U$, $0 \leq t \leq T$, при условиях $F_i[u(\cdot)]=0$, $i=1, 2, \dots, m$, $G[x(t)] \leq 0$ при всех t . Для F_i примем конкретную формулу типа

$$F_i[u(\cdot)] \equiv \int_0^T \Phi^{(i)}[x(t), u(t)] dt.$$

Предположение теоремы состоит в следующем: существует функция $\psi(t)$, решение краевой задачи

$$\begin{aligned} \frac{d\psi}{dt} + f_x^*[t] \psi &= - \sum_{i=0}^m g_i \Phi_x^{(i)}[t] + G_x[t] \frac{d\sigma}{dt}; \\ \Gamma_x^* \psi &= 0, \end{aligned} \quad (1)$$

причем $g_0 = -1$, $\sigma(t)$ — монотонная функция с точками роста на множестве $M = \{t: G[x(t)] = 0\}$.

Образованная из $x(t)$ и $\psi(t)$ функция

$$H[x(t), \psi(t), u] \equiv \sum_{i=0}^m g_i \Phi^{(i)}[x(t), u] + (\psi(t), f[x(t), u])$$

в области U при всех $t \in [0, T]$ достигает максимума в точке $u(t)$.

Рассмотрим теперь малую вариацию управления $\delta u(\cdot)$, удовлетворяющую следующим условиям:

- $u(t) + \delta u(t) \in U$ при всех t ,
- $\delta F_i[\delta u(\cdot)] = 0$, $i = 1, 2, \dots, m$,
- $G_x[x(t)] \delta x(t) \leq 0$, $t \in M$.

Здесь $\delta x(t)$ — вариация фазовой траектории, являющаяся следствием вариации управления $\delta u(\cdot)$; связь между ними дает уравнение в вариациях

$$\begin{aligned} \frac{d\delta x}{dt} &= f_x[t] \delta x + f_u[t] \delta u; \\ \Gamma_x \delta x &= 0. \end{aligned} \quad (2)$$

В остальном вариация $\delta u(\cdot)$ произвольна. Тогда $\delta F_0[\delta u(\cdot)] \geq 0$. В этом состоит утверждение о стационарности подобной траектории.

Доказательство. Из принципа максимума следует:

$$H_u[x(t), \psi(t), u(t)] \delta u \leq 0 \quad \text{для всех } t, \delta u \in K_u.$$

Интегрируя это соотношение, получим

$$\begin{aligned} 0 &\geq \int_0^T H_u[x(t), \psi(t), u(t)] \delta u(t) dt = \\ &= \int_0^T \left\{ \sum_i g_i \Phi_u^{(i)}[x(t), \psi(t), u(t)] \delta u(t) + (\psi(t), f_u[x(t), u(t)]) \delta u(t) \right\} dt \stackrel{(2)}{=} \\ &= \int_0^T \left\{ \sum_{i=0}^m g_i \Phi_u^{(i)} \delta u + \left(\psi, \left(\frac{d}{dt} - f_x \right) \delta x \right) \right\} dt = \\ &= \int_0^T \left\{ \sum_{i=0}^m g_i \Phi_u^{(i)} \delta u + \left(\delta x, \left(\frac{d}{dt} - f_x \right)^* \psi \right) \right\} dt \stackrel{(1)}{=} \\ &= \int_0^T \left\{ \sum_{i=0}^m g_i \Phi_u^{(i)} \delta u + \sum_{i=0}^m g_i \Phi_x^{(i)} \delta x - \frac{d\sigma}{dt} G_x[t] \delta x \right\} dt = \\ &= \int_0^T \left\{ \sum_{i=0}^m g_i [\Phi_x^{(i)} \delta x + \Phi_u^{(i)} \delta u] - \frac{d\sigma}{dt} G_x \delta x \right\} dt \stackrel{*}{=} \\ &= \sum_{i=0}^m g_i \delta F_i - \int_M \frac{d\sigma}{dt} G_x[t] \delta x(t) dt = -\delta F_0 - \int_M \frac{d\sigma}{dt} G_x[x(t)] \delta x(t) dt \leq 0. \end{aligned}$$

И окончательно

$$\delta F_0[\delta u(\cdot)] \geq - \int_M \frac{d\sigma}{dt} G_x[t] \delta x(t) dt \geq 0.$$

*) Используем то, что $\int_0^T \{ \Phi_x^{(i)} \delta x + \Phi_u^{(i)} \delta u \} dt = \delta F_i$, $d\sigma/dt = 0$ вне M , $d\sigma/dt \geq 0$ на M и, по предположению, $G_x \delta x(t) \leq 0$ на M , $\delta F_i = 0$, $i = 1, 2, \dots, m$, $g_0 = -1$.

Этим доказательство заканчивается. По этой же в точности схеме оно может быть проведено и для конечных вариаций управления на множестве малой меры, и для задачи с параметрами, и для остальных обобщений, рассмотренных в § 7.

§ 10. Вопросы существования решений

В этом параграфе будут рассмотрены некоторые чисто теоретические аспекты вариационной задачи; однако они имеют и практическое значение; ясное понимание их необходимо при конструировании численных методов, рассчитанных на достаточно широкий класс прикладных задач.

1. Существование оптимального управления. Итак, ищется управление $u(\cdot)$, минимизирующее значение $F_0[u(\cdot)]$ при условиях $F_i[u(\cdot)] = 0, i = 1, 2, \dots, m$, где F — функционалы, определенные на траектории системы

$$\dot{x} = f(x, u), \quad \Gamma(x) = 0; \quad u(t) \in U, \quad 0 \leq t \leq T.$$

Вопрос о существовании оптимального управления мы разберем, рассмотрев возможные формы несуществования его.

1. Первый банальный случай несуществования. Пусть среди всех U -допустимых измеримых функций ($u(t) \in U$ при всех t) нет такой, которая обеспечивала бы выполнение условий $F_i[u(\cdot)] = 0, i = 1, 2, \dots, m$, при каком угодно значении $F_0[u(\cdot)]$. В этом случае нет вариационной задачи и говорить не о чем. Эта возможность в теории оптимального управления отвергается, считается (и это соответствует положению дел при решении прикладных задач), что по крайней мере одно допустимое *) управление существует.

2. Второй банальный случай несуществования. Пусть существует последовательность управлений $u^{(k)}(\cdot)$, допустимых и доставляющих F_0 сколь угодно малые (алгебраически) значения

$$F_0[u^{(k)}(\cdot)] \rightarrow -\infty, \quad k \rightarrow \infty.$$

Пусть при этом ни о каком содержательно приемлемом пределе $\{u(\cdot), x(\cdot)\}$, к которому сходятся траектории $\{u^{(k)}(\cdot), x^{(k)}(\cdot)\}$, говорить нельзя. Такая ситуация свидетельствует о плохой постановке задачи и не рассматривается. В теоретическом анализе она предполагается места не имеющей, что не вызывает никаких возражений и с «прикладной» точки зрения.

*) Напомним, что допустимым управлением $u(t) \in U$ называют такое, которое обеспечивает выполнение условий

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m.$$

3. Случай отсутствия оптимального управления среди измеримых функций $u(t)$. Это уже достаточно интересный случай несуществования. Пусть имеется точная нижняя грань значений $F_0[u(\cdot)]$ в классе допустимых управлений $u(\cdot)$:

$$F_0 = \inf_{u(\cdot)} F_0[u(\cdot)].$$

Тогда существует и минимизирующая последовательность $u^{(k)}(\cdot)$:

$$\lim_{k \rightarrow \infty} F_0[u^{(k)}(\cdot)] = F_0.$$

Естественно возникает вопрос: нельзя ли из последовательности траекторий $\{u^{(k)}(\cdot), x^{(k)}(\cdot)\}$ выделить сходящуюся в том или ином смысле подпоследовательность, и предел последней $\{u(\cdot), x(\cdot)\}$ считать решением вариационной задачи? Ответ оказывается разным для $u(t)$ и для фазовой траектории $x(t)$.

Типичным является существование предельной фазовой траектории $\tilde{x}(t)$, причем сходимость оказывается равномерной: существует $\tilde{x}(t)$ такая, что *)

$$\max_t \|x^{(k)}(t) - \tilde{x}(t)\| \rightarrow 0 \quad \text{при} \quad k \rightarrow \infty.$$

Этот факт устанавливается при весьма общих предположениях, выполняющихся в большинстве прикладных задач: пусть для любого управления $u(t) \in U$, $t \in [0, T]$, определяемая краевой задачей фазовая траектория ограничена: $\|x(t)\| \leq C$; пусть при этом все возможные значения $\|f(x, u)\| \leq C_1$, т. е. $\|dx/dt\| \leq C_1$. Множество таких функций $x^{(k)}(\cdot)$ удовлетворяет условиям известной теоремы Арцела (критерий компактности в $C^0[a, b]$), и из любой бесконечной совокупности таких функций можно выделить равномерно сходящуюся подпоследовательность, причем предельная функция $\tilde{x}(t)$ будет ограниченной, почти всюду дифференцируемой, и

$$\|\tilde{x}(t)\| \leq C, \quad \|\dot{\tilde{x}}(t)\| \leq C_1.$$

Для последовательности функций $u^{(k)}(\cdot)$ дело не так просто, как это хорошо показывает следующий пример:

Найти $\max_{u(\cdot)} \int_0^3 [x(t) + u^2(t)] dt$ при условиях:

- 1) $\dot{x} = u; \quad x(0) = 0; \quad x(3) = 0,$
- 2) $|u(t)| \leq 1,$
- 3) $x(t) \leq 1.$

*) Ради простоты мы не вводим обозначений для подпоследовательности.

Задача легко анализируется: в самом деле, почти очевидна конструкция максимизирующей последовательности управлений

$$u^{(k)}(t) = \begin{cases} 1 & \text{при } 0 \leq t < 1, \\ -1 & \text{при } 2 < t \leq 3, \end{cases}$$

а на интервале $[1, 2]$ $u^{(k)}(t)$ строится так: $[1, 2]$ разбивается на $2k$ равных частей, на нечетных частях $u(t) = -1$, на четных

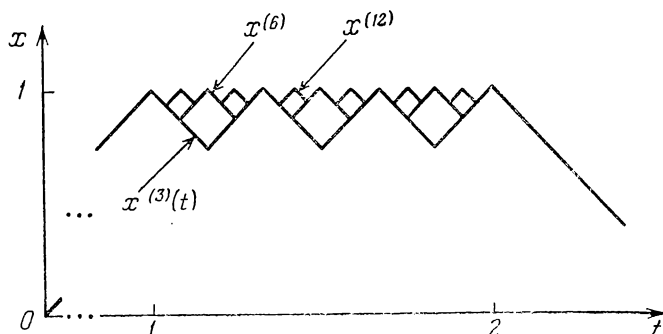


Рис. 7.

$u(t) = 1$ (см. рис. 7, на котором изображены функции $x^{(k)}(t)$ для $k = 3, 6, 12$). Ясно, что $F_0[u^{(k)}(\cdot)] \rightarrow 5$, $x^{(k)}(t) \rightarrow \tilde{x}(t)$, где

$$\tilde{x}(t) = \begin{cases} t, & 0 \leq t \leq 1, \\ 1, & 1 \leq t \leq 2, \\ 3 - t, & 2 \leq t \leq 3. \end{cases}$$

Никакого предела для функций $u^{(k)}(\cdot)$ нет. Более того, предельная функция $\tilde{x}(t)$ определяется допустимым управлением

$$\tilde{u}(t) = \begin{cases} 1, & 0 \leq t < 1, \\ 0, & 1 \leq t \leq 2, \\ -1, & 2 < t < 3, \end{cases}$$

однако траектория $\{\tilde{u}(\cdot), \tilde{x}(\cdot)\}$ не оптимальна; $F_0[\tilde{u}(\cdot)] = 4$. Заметим, что можно усложнить ситуацию, определив область U не условием $|u| \leq 1$, а считая ее состоящей только из двух точек: 1 и -1 (т. е. $u = 1$ или $u = -1$). В этом случае предельная траектория $\tilde{x}(t)$ не соответствовала бы никакому допустимому управлению.

Эти вопросы связаны с замыканием того множества элементов функционального пространства, на котором определены входящие в постановку задачи функционалы: известно, что непрерывная ограниченная функция на замкнутом ограниченном мно-

жестве достигает минимального значения; поэтому вопросы замыкания самым тесным образом связаны с вопросом существования экстремальной точки. Сама же операция замыкания формулируется в терминах определенной нормы и состоит в том, что сходящиеся в нужном по содержанию задачи смысле последовательности элементов первоначального пространства объявляются элементами нового, пополненного пространства; последнее оказывается уже замкнутым. По существу, это означает, что сходящуюся в определенном смысле минимизирующую последовательность траекторий $\{u^{(k)}(\cdot), x^{(k)}(\cdot)\}$ мы объявляем решением задачи, хотя возникающий при этом предельный элемент и не обладает всеми свойствами элементов первоначальной постановки задачи.

Для задач оптимального управления естественным является замыкание, порождаемое следующим определением сходящейся последовательности траекторий $\{u^{(k)}(\cdot), x^{(k)}(\cdot)\}$, где $u^{(k)}(\cdot)$ — измеримая функция, а $x^{(k)}(\cdot)$ — порожденная ею фазовая траектория. Такую последовательность называют *сходящейся в себе*, если при любом $\varepsilon > 0$ для всех достаточно больших чисел $k, q > K(\varepsilon)$ выполнены соотношения:

- 1) $\max_t \|x^{(k)}(t) - x^{(q)}(t)\| \leq \varepsilon,$
- 2) $|F_i[u^{(k)}(\cdot)] - F_i[u^{(q)}(\cdot)]| \leq \varepsilon, \quad i = 0, 1, \dots, m.$

Следствием этого является существование предельной фазовой траектории $\tilde{x}(t)$ и предельных значений всех входящих в постановку задачи функционалов. Таким образом, сходящейся последовательности траекторий управляемой системы соответствуют некоторая фазовая траектория $\tilde{x}(t)$ (она почти всюду имеет производную и удовлетворяет краевым условиям задачи $\Gamma(x)=0$) и значения функционалов \tilde{F}_i ($i=0, 1, \dots, m$). Рассмотренный выше пример показал, что мы можем столкнуться, по крайней мере, с тремя ситуациями:

1. Траектория $\tilde{x}(t)$ порождена некоторым измеримым управлением $\tilde{u}(\cdot)$, и при этом

$$F_i[\tilde{u}(\cdot)] = \tilde{F}_i, \quad i = 0, 1, \dots, m.$$

2. Траектория $\tilde{x}(t)$ порождена измеримым управлением $\tilde{u}(\cdot)$, но

$$F_i[u(\cdot)] \neq \tilde{F}_i \text{ хотя бы для одного } i.$$

3. Траектория $\tilde{x}(t)$ не может быть получена никаким U -допустимым управлением $u(\cdot)$.

В любом случае предельный комплекс $\{\tilde{x}(\cdot), \tilde{F}_0, \tilde{F}_1, \dots, \tilde{F}_m\}$ имеет содержательный смысл: существует «нормальная» траектория $\{u(\cdot), x(\cdot)\}$, сколь угодно мало отличающаяся от предельного комплекса по основным характеристикам управления.

Применим эти соображения к минимизирующей последовательности траекторий $\{u^{(k)}(\cdot), x^{(k)}(\cdot)\}$:

$$\begin{aligned} x^{(k)}(t) &\rightarrow \tilde{x}(t), \quad \|x^{(k)}(t)\| \leq C; \quad \|\tilde{x}^{(k)}(\cdot)\| \leq C_1, \\ F_i[u^{(k)}(\cdot)] &= 0, \text{ или } F_i[u^{(k)}(\cdot)] \rightarrow 0, \quad i = 1, 2, \dots, m, \\ F_0[u^{(k)}(\cdot)] &\rightarrow \inf_{u(\cdot)} F_0[u(\cdot)], \quad k \rightarrow \infty. \end{aligned}$$

По теореме Арцела из такой последовательности можно выбрать сходящуюся подпоследовательность, и предельный комплекс $\{\tilde{x}(\cdot), \tilde{F}_0, \tilde{F}_1, \dots, \tilde{F}_m\}$ считать решением вариационной задачи. Однако этим дело не кончается: неясен вопрос о необходимых условиях типа принципа максимума для этого решения; ведь все выкладки § 5 были проведены на некоторой «обычной» траектории $\{u(\cdot), x(\cdot)\}$. Это очень неприятное обстоятельство прежде всего для тех численных методов, которые основаны на прямом использовании принципа максимума. На первый взгляд оно не очень существенно для большинства приближенных методов решения вариационных задач, которые принципа максимума не используют (точнее, используют его негативную формулировку), а состоят в построении минимизирующей последовательности управлений. Однако это не так, и позже мы дадим более подробные разъяснения по этому поводу (см. стр. 197).

Поэтому очень интересен вопрос о точном описании всех предельных комплексов $\{\tilde{x}(\cdot), \tilde{F}_0, \dots, \tilde{F}_m\}$ и о выделении класса задач, в которых все предельные комплексы являются «нормальными» траекториями $\{\tilde{u}(\cdot), \tilde{x}(\cdot)\}$, т. е. когда реализуется только первая из трех описанных выше ситуаций. Оба вопроса были решены А. Ф. Филипповым; мы сформулируем и разъясним его результат, адресуя читателя к [97] за подробным доказательством. Однако сначала заметим, что достаточно ограничиться функционалами типа

$$F[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt. \quad (1)$$

С функционалами, зависящими только от $x(\cdot)$ (в том смысле, как это разъяснено на стр. 65, вопрос ясен: они определены и на предельной функции $\tilde{x}(\cdot)$, и $F[\tilde{x}(\cdot)] = \tilde{F}$. Для дальнейшего удобно расширить систему $\dot{x} = f(x, u)$, присоединив к ней уравнения типа

$$\begin{aligned} dx^{n+1}/dt &= \Phi_0(x, u), \quad x^{n+1}(0) = 0, \\ dx^{n+m+1}/dt &= \Phi_m(x, u), \quad x^{n+m+1}(0) = 0. \end{aligned}$$

(Тогда, например, $F_0[u(\cdot)] = x^{n+1}(T)$.) Новую систему из $(n + m + 1)$ уравнений запишем в прежней форме:

$$dx/dt = f(x, u), \quad \Gamma(x) = 0,$$

и рассмотрим предельную функцию $\tilde{x}(\cdot)$.

Теорема Филиппова. *Предельная траектория $\tilde{x}(\cdot)$ является абсолютно непрерывной функцией, имеющей почти при всех t производную $d\tilde{x}/dt$, причем эта производная является измеримой функцией и удовлетворяет условию*

$$d\tilde{x}/dt \in \text{conv } f(x, U). \quad (2)$$

Здесь $f(x, U)$ есть совокупность точек $\{f(x, u)\}_{u \in U}$, а $\text{conv } f(x, U)$ — выпуклая оболочка $f(x, U)$.

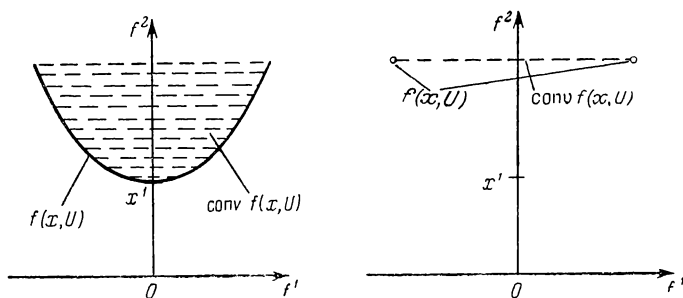


Рис. 8.

Следствие. Если $f(x, U)$ — выпуклое множество, то предельный элемент $\tilde{x}(\cdot)$ является решением уравнения $d\tilde{x}/dt = f(\tilde{x}, \tilde{u}(t))$ с некоторым измеримым управлением (U -допустимым, разумеется). Сначала рассмотрим с этой точки зрения пример. Расширенная система имеет вид

$$\begin{aligned} dx^1/dt &= u; & x^1(0) &= 0, \\ dx^2/dt &= x^1 + (u)^2; & x^2(0) &= 0. \end{aligned}$$

Область $f(x, U)$ изображена на рисунке 8 для двух случаев: 1) $|u| \leq 1$ и 2) $u=1$ или -1 . В том и другом случае множество $f(x, U)$ невыпукло и штриховкой показана его выпуклая оболочка $\text{conv } f(x, U)$. Изображенное на рис. 7*) решение на интервале $1 \leq t \leq 2$ удовлетворяет уравнению

$$\begin{Bmatrix} \dot{x}^1 \\ \dot{x}^2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ x^1 + 1 \end{Bmatrix} \in \text{conv } f(x, U); \quad \begin{Bmatrix} \dot{x}^1 \\ \dot{x}^2 \end{Bmatrix} \notin f(x, U).$$

Теперь поясним, почему в результате замыкания множества траекторий $\{u(\cdot), x(\cdot)\}$, порожденного всеми измеримыми функциями $u(\cdot)$, появляются функции $x(t)$, удовлетворяющие включению

$$dx/dt \in \text{conv } f(x, U), \quad 0 \leq t \leq T.$$

*) На рис. 7 $x(t)$ есть координата $x^1(t)$ расширенной системы.

Разобьем интервал $[0, T]$ на малые интервалы длиной $\tau = T/N$; на каждом интервале длиной τ определим кусочно постоянную функцию $u(t)$ следующим образом: для произвольных чисел $\alpha_1, \alpha_2, \dots$

$\dots, \alpha_l \geq 0, \sum_{k=1}^l \alpha_k = 1$, разобьем интервал длиной τ на l частей длиной $\alpha_k \tau$; пусть теперь $u_k, k = 1, 2, \dots, l$, — произвольные точки из U ; определим $u(t) = u_k$ на k -й части интервала τ . Далее,

$$\begin{aligned} x(t' + \tau) &= x(t') + \int_{t'}^{t' + \tau} f[x(t), u(t)] dt = \\ &= \int_{t'}^{t' + \tau} f[x(t'), u(t)] dt + O(\tau^2) = \tau \sum_{k=1}^l f[x(t'), u_k] \alpha_k + O(\tau^2). \end{aligned}$$

Но $\sum_{k=1}^l f(x, u_k) \alpha_k \in \text{conv } f(x, U)$, таким образом, любая абсолютно непрерывная функция $x(t)$, имеющая почти всюду производную — измеримую функцию $\dot{x}(t)$, удовлетворяющую включению (2), может быть с любой степенью точности аппроксимирована решением уравнения $\dot{x} = f[x, u(t)]$ с кусочно постоянной функцией $u(t)$.

Аналогично может быть проверен и обратный факт: любое решение системы $\dot{x} = f[x, u(t)]$ с измеримой U -допустимой функцией $u(t)$ с любой точностью аппроксимируется решением включения (2). В самом деле,

$$x(t' + \tau) = x(t') + \int_{t'}^{t' + \tau} f[x(t'), u(t)] dt + O(\tau^2), \quad u(t) \in U,$$

и нетрудно показать, что

$$\frac{1}{\tau} \int_{t'}^{t' + \tau} f[x(t'), u(t)] dt \in \text{conv } f[x(t'), U]$$

при любой измеримой U -допустимой $u(t)$.

2. Скользящие режимы. Итак, если для расширенной системы множество $f(x, U)$ оказывается невыпуклым, есть основания ожидать отсутствия оптимальной траектории $\{u(\cdot), x(\cdot)\}$; оптимальной может оказаться траектория включения $\dot{x} \in \text{conv } f(x, U)$. В. Ф. Кротов, видимо, одним из первых обратил внимание на то, что эта ситуация отнюдь не является продуктом присущего чистой математике стремления рассмотреть и проанализировать все возможные варианты. Подобные предельные объекты, оказывается, появляются и в прикладных задачах оптимального управления. Они получили специальное название «скользящие режимы» и потребовались дополнительные исследования для вывода необходимых условий оптимальности.

Р. В. Гамкрелидзе предложил следующий прием сведения задачи с невыпуклым множеством $f(x, U)$ к задаче с выпуклым. В сущности, это есть способ стандартного аналитического описания дифференциального включения $\dot{x} \in \text{conv } f(x, U)$. В теории выпуклых множеств известен следующий факт: любую точку выпуклой оболочки $f(x, U)$ можно получить в виде

$$f = \sum_{k=1}^{n+1} \alpha_k f(x, u_k); \quad \alpha_k \geq 0; \quad \sum_{k=1}^{n+1} \alpha_k = 1, \quad u_k \in U. \quad (3)$$

Здесь n — размерность пространства, u_k — некоторые точки, $u_k \in U$ (разумеется, не фиксированные; важно то, что этих точек достаточно не более $(n+1)$ -й; напомним также, что n — размерность расширенной системы, равная сумме размерности исходного фазового пространства и числа функционалов, явно зависящих от $u(\cdot)$).

Введем вместо исходного управления $u(\cdot)$ новые управляющие функции: $u^{(k)}(\cdot)$, $\alpha_k(\cdot)$, $k=1, 2, \dots, n+1$ ($u^{(k)}$ — вектор, α_k — скаляр). Систему (расширенную) запишем в виде

$$\frac{dx}{dt} = \sum_{k=1}^{n+1} \alpha_k(t) f[x, u^{(k)}(t)].$$

Геометрическое ограничение приобретает вид

$$\begin{aligned} u^{(k)}(t) &\in U, & k=1, 2, \dots, n+1, \\ \alpha_k(t) &\geq 0, & k=1, 2, \dots, n+1, \\ \alpha_1(t) + \alpha_2(t) + \dots + \alpha_{n+1}(t) &= 1. \end{aligned}$$

Заметим, что размерность r^* новой управляющей функции связана с n и с размерностью исходного управления r соотношением $r^* = (n+1)(r+1)$. Теперь можно без существенного ограничения общности изучать экстремальные траектории, считая их порожденными достаточно простыми, например, кусочно гладкими функциями $u^{(k)}(\cdot)$, $\alpha_k(\cdot)$.

Отметим здесь же два неприятных обстоятельства: повышение размерности нового управления и формальную возможность тривиальной неединственности решения: ведь представление точек выпуклой оболочки $\text{conv } f(x, U)$ в виде (3) — неединственно. Правда, в оптимальном управлении нас интересуют не все точки $\text{conv } f(x, U)$, а лишь граничные; это обстоятельство сужает возможную неединственность (для областей, изображенных на рис. 8, граничные точки имеют единственное представление в виде (3)), но не исключает ее совсем.

Одним из аспектов классического вариационного исчисления было исследование таких минимальных расширений первоначального множества дифференцируемых функций (а именно на них

определен типичный классический функционал $\int_{-1}^1 \Phi[x, \dot{x}] dt$ и самого способа вычисления минимизируемого функционала, чтобы на расширенном множестве «траекторий сравнения» решение вариационной задачи существовало. Следующий пример Вейерштрасса хорошо иллюстрирует сущность подобных исследований:

$$\min \int_{-1}^1 t^2 \dot{x}^2 dt, \quad x(-1) = -1; \\ x(1) = 1.$$

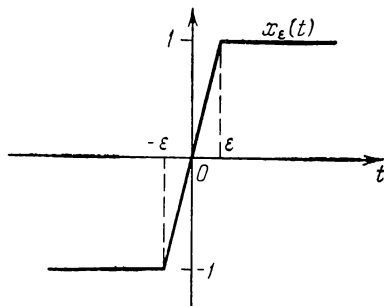


Рис. 9.

Почти очевидно, что представленная на рис. 9 функция $x_\epsilon(t)$ при $\epsilon \rightarrow 0$ есть элемент минимизирующей последовательности.

В самом деле,

$$\int_{-1}^1 t^2 \dot{x}^2 dt = \int_{-\epsilon}^{\epsilon} t^2 \left(\frac{1}{\epsilon}\right)^2 dt = O(\epsilon) \rightarrow 0$$

при $\epsilon \rightarrow 0$. В то же время

$$\inf \int_{-1}^1 t^2 \dot{x}^2 dt \geq 0.$$

Таким образом, предел последовательности $\lim_{\epsilon \rightarrow 0} x_\epsilon(t) = \text{sign } t$ есть разрывная функция.

Упомянутое расширение первоначального множества дифференцируемых функций состоит в присоединении к нему и разрывных; минимальность этого расширения достигается тем, что не все разрывные функции считаются допустимыми; требуется выполнение определенных соотношений в точке разрыва. В теории оптимального управления, точнее, в той ее части, которая ориентирована на разработку приближенных методов решения прикладных задач, исследования подобного рода не очень интересны. Это связано с тем, что сама форма уравнений

$$\frac{dx}{dt} = f(x, u), \quad u \in U,$$

практически во всех прикладных задачах обеспечивает ограниченность производной \dot{x} : $\|\dot{x}\| = \|f(x, u)\| \leq C$ при всех интересующих нас значениях x .

Сказанное выше находится в некотором противоречии с теми примерами задач оптимального управления, которые будут в дальнейшем решены. В некоторых из них управления содержат особенности типа δ -функции, а фазовые траектории $x(t)$ оказываются разрывными. Не вдаваясь в подробности, рассмотрим причины, которые привели к таким странным решениям.

1. В задаче о вертикальном подъеме ракеты (§ 29) оптимальное управление содержит δ -функцию с полюсом в точке $t=0$. Ее появление связано с тем, что в постановку задачи включено лишь имеющее физический смысл ограничение $u(t) \geq 0$. Более реалистическая постановка этой задачи должна включать и «техническое» условие $u(t) \leq U$ (ограничение мощности двигателя), после чего δ -функция в решении исчезает.

2. В § 35 решается задача, в которой оптимальное управление содержит две δ -функции (а $x(t)$ — две точки разрыва). Однако задача не имеет прикладного смысла и является искусственно сконструированным тестом.

3. В задаче о стабилизации спутника (§ 34) оптимальное управление также содержит δ -функции. Их появление связано с тем, что постановка задачи не включает в себя естественное ограничение $|u| \leq U$, имеющее смысл ограничения мощности реактивных двигателей.

4. Содержат δ -функции и оптимальные управления в решении задачи об остановке реактора (§ 36) и в задаче оптимизации характеристик ядерного реактора (§ 38). Их появление связано с применением искусственного приема: функция $u(t)$, являющаяся управлением в первичной постановке задачи, была превращена в фазовую координату, связанную с новым управлением $v(t)$ уравнением $du/dt = v(t)$.

Для того чтобы получить разрывное управление $u(t)$, нужно иметь δ -функцию в $v(t)$. Однако в задаче § 38 был применен прием, позволивший получить разрывное $u(t)$ без δ -функций.

Таким образом, если придерживаться сугубо «прикладной» точки зрения, можно считать появление δ -функций в искомым оптимальных управлениях исключенным. Однако с тех же позиций можно запретить и появление разрывов в $u(t)$. Обычно подобный запрет аргументируется примерно таким образом: управление реальной системой, т. е. изменение со временем t входящей в уравнения движения функции $u(t)$, осуществляется некоторой аппаратурой, имеющей конечное время срабатывания, и фактически реализовать разрыв в $u(t)$ нельзя — он всегда в той или иной мере «размазан» во времени. Это не очень удачная аргументация. Она не учитывает, по крайней мере, двух обстоятельств. Во-первых, независимый аргумент t задачи не всегда имеет физический смысл времени. В задачах, например, о выборе оптимальных композиций защиты (§ 33), об оптимизации реактора (§ 38) t

не является временем, вопроса о «времени» срабатывания нет, и разрывные $u(t)$ являются вполне естественными и фактически реализуемыми (например, защита может состоять из чередующихся слоев разных веществ).

Более существенным является другое соображение: задачи оптимального управления ставятся для достаточно упрощенных моделей реальных инженерных объектов, и использование в этих моделях таких «чисто математических» изобретений, как разрывные функции и δ -функции, связано с наличием в задаче малых (или больших) параметров. Так, в задаче о подъеме ракеты (§ 29), если техническое ограничение $u(t) \leq U^+$ таково, что $U^+T \gg 1$ (T — характерное время в задаче), то и модель с $U^+ = \infty$ (приводящая к решению с δ -функцией) оказывается приемлемой (с тем большими основаниями, чем больше величина U^+T). Точно так же, если время срабатывания реализующей управление аппаратуры τ таково, что $\tau \ll T$ (T — характерное время задачи), то и математическая идеализация с разрывным управлением $u(t)$ оказывается естественной. И с точки зрения трудности численного решения задач оптимального управления, как мы увидим в дальнейшем, важны не формальные словесные характеристики искомых функций, например, «непрерывность», а более четкое и содержательное выделение классов функций. Для вычислителя разница между классом функций, удовлетворяющих условию Липшица

$$|u(t') - u(t)| \leq C |t' - t|$$

и классом произвольных (измеримых) функций тем меньше, чем больше величина CT/U (T — промежуток времени, на котором нас интересует функция $u(t)$, а U — характерная величина $u(t)$ в данной задаче).

К а н т о р о в а л е с т н и ц а. Выше было отмечено, что любая вариационная задача в классической постановке легко может быть сформулирована как задача оптимального управления. Формально обе задачи оказываются эквивалентными, однако есть между ними и некоторая, так сказать, «идеологическая», разница. Пояснить ее лучше всего, вспомнив интересный пример (он поучителен и сам по себе).

К л а с с и ч е с к и й в а р и а н т з а д а ч и: найти

$$\min_{x(\cdot)} \int_0^1 \sqrt{1 + \dot{x}^2} dt$$

при условиях $x(0) = 0$; $x(1) = 1$.

Естественным решением этой задачи является функция $x(t) \equiv t$,

при этом $\int_0^1 \sqrt{1+x^2} dt = \sqrt{2}$. Однако постановка задачи не полна: не указан класс функций $x(t)$, среди которых ищется минимум $\int_0^1 \sqrt{1+x^2} dt$. Кантором был построен пример функции $y(t)$ непрерывной, монотонно растущей, почти всюду имеющей производную, и эта производная почти всюду равна нулю! Таким образом, $\int_0^1 \sqrt{1+y^2} dt = 1$. Разумеется, эта функция $y(t)$ относится к числу математических «монстров». С прикладной точки зрения основной причиной отвергать подобные «решения» является следующая: не существует такой аппроксимирующей последовательности функций $y^{(k)}(t)$, являющихся «нормальными» (т. е., например, непрерывными и имеющими кусочно непрерывную производную, ограниченную постоянной C_k , пусть своей для каждого k , и пусть даже $C_k \rightarrow \infty$ при $k \rightarrow \infty$), для которой

$$\lim_{k \rightarrow \infty} \int_0^1 \sqrt{1 + [\dot{y}^{(k)}]^2} dt = 1.$$

Если бы такая последовательность существовала, функцию $y(t)$ следовало бы считать полезной и с прикладной точки зрения. С чисто математической точки зрения дефектом этой функции $y(t)$ (канторовой лестницы) является то, что она не является первообразной для своей производной *).

Задача оптимального управления: найти

$$\min_{u(\cdot)} \int_0^1 \sqrt{1+u^2} dt$$

на траектории системы

$$\dot{x} = u; \quad x(0) = 0; \quad x(1) = 1.$$

Эта постановка уже в достаточно четкой форме предполагает, что $x(t)$ должна быть первообразной для $u(t)$. А если добавить еще условие $|u(t)| \leq U^+$, то появление столь странных объектов, как канторова лестница, оказывается полностью исключенным.

*) Канторова лестница строится так: отрезок $[0, 1]$ разбивается на три равные части, в средней полагается $y(t) = 0,5$. Левый (правый) пустой отрезок в свою очередь разбивается на три части и в средней полагается $y(t) = 0,25$ ($0,75$) и т. д. Предельная функция $y(t)$ непрерывна, но не абсолютно непрерывна.

Скольльзящие режимы и прикладные задачи. Выше был рассмотрен характерный пример вариационной задачи, в которой экстремум достигается на скольльзящем режиме. Речь идет о следующей ситуации: строится оптимизирующая последовательность траекторий и рассматривается ее предел. Оказалось, что фазовые компоненты этой последовательности имеют в качестве предела достаточно гладкую функцию. Но соответствующие члены последовательности управлений (или, если угодно, производных фазовых траекторий) естественного предела не имеют. Аналогичные примеры строились и в классическом вариационном исчислении. Например, задача отыскания

$$\min_{x(\cdot)} \int_0^1 (x^2 - \dot{x}^2) dt \text{ при условиях } x(0) = x(1) = 0,$$

также приводит к «решению» типа скольльзящего режима.

Возникающие в связи с подобными ситуациями сложные теоретические вопросы являются предметом изучения большого числа математических работ. Тому, кто занимается приближенным решением задач оптимального управления, нужно иметь какую-то точку зрения на эти исследования, так же как и на многие другие исследования весьма далеких и абстрактных обобщений вариационных задач. Нужно решить, с чем связано это дальнейшее развитие теории: с необходимостью включить в нее какой-то новый класс прикладных задач, или с характерным для современной математики стремлением к возможно большей общности, к ослаблению предположений, при которых доказываются те или иные теоремы. В первом случае следует соответствующим образом модернизировать вычислительные методы или создать новые с тем, чтобы можно было находить приближенные решения и для нового класса задач. Во втором случае, в принципе, можно, признав эти обобщения не имеющими (в настоящее время, во всяком случае) отношения к прикладным задачам, не усложнять и без того не простую задачу приближенного решения стремлением не отстать от чисто теоретических обобщений. Ведь в конце концов в приближенном решении нуждаются прежде всего и в основном задачи, имеющие прикладное значение, и специалисту по прикладной математике естественно ограничиться (при реализации приближенных методов) тем уровнем теории, которым охватываются типичные прикладные задачи. Сразу же возникает вопрос: а что такое «класс прикладных задач», как его можно охарактеризовать? Видимо, ответить на этот вопрос можно, только проанализировав возможно большее число вариационных задач, поставленных физиками, химиками, инженерами, специалистами по космонавтике и другими учеными, имеющими дело непосредственно с объектами реального мира. Разумеется, этот материал даст ответ, связанный с се-

годняшним уровнем математизации естественных наук. Не исключено, что через несколько лет положение изменится, и разделы теории, относящиеся сегодня к числу абстрактных, перейдут в ряд прикладных.

С этой точки зрения имеет смысл разобраться в прикладном значении теории, связанной со скользящими режимами, и решить, следует ли строить приближенные методы в расчете и на этот случай, или можно рассчитывать на то, что в прикладной задаче появление скользящего режима маловероятно. Дело в том, что, как это будет подробно объяснено в § 23, желая серьезно решать задачи со скользящими режимами, вычислитель должен будет существенно усложнить алгоритм приближенного решения. На это имеет смысл идти лишь в том случае, когда это действительно нужно.

Разбираясь в этом вопросе, нужно серьезно обсудить теоретические взгляды, изложенные в монографиях Янга [101] и Кротова и Гурмана [39]. В них подчеркивается практическая необходимость изучения вырожденных решений вариационных задач (обобщенных кривых, скользящих режимов и т. д.) и разрабатывается соответствующий достаточно сложный математический аппарат.

В монографии Янга специально подчеркивается опасность использования «наивного» вариационного исчисления (уравнения Эйлера и предположения о том, что решение задачи существует и является достаточно простой, гладкой функцией): «В действительности, как мы увидим, . . . метод Эйлера имеет самые серьезные недостатки как в теории, так и на практике» ([101], стр. 23). Но не следует забывать, что метод Эйлера самым активным образом используется физиками, механиками и инженерами (как в теории, так и на практике) вот уже около двухсот лет в задачах не специально сконструированных изобретательным математиком, а естественно возникших в приложениях. И не так-то просто в этой огромной практике найти пример, когда бы этот «наивный» подход привел к серьезной ошибке, причем такой, исправление которой было бы возможно лишь с использованием тонких обобщений, рассматриваемых в [101]. Во всяком случае, в [101] таких примеров нет, хотя эта книга изобилует беллетризованными иллюстрациями теоретических ситуаций, «примерами из жизни». Этим примерам самым существенным образом не хватает реальной основы, т. е. настоящей, четко поставленной вариационной задачи, связанной с описываемой жизненной ситуацией, причем такой задачи, в которой эйлеров подход привел бы к серьезному просчету, а «скрытое решение» (это примерно то же самое, что и скользящий режим) давало бы правильный ответ. Без таких задач многочисленные «примеры из практики» в [101] выглядят не очень убедительно.

Практическую актуальность вырожденных решений подчеркивают и авторы [39]: «термин «вырожденный», применяющийся в математике для обозначения редких ситуаций, имеет здесь иной

смысл. В самом деле, из определения вырожденного режима следует, что оптимальный режим будет заведомо невырожденным лишь тогда, когда функция Гамильтона $H(x, \phi, u)$ строго выпукла на множестве U . Но это гораздо более редкая ситуация, чем противоположная, в которой вырожденный режим нельзя исключать из рассмотрения в процессе исследования; это подтверждается на практике. Например, решения типичных задач оптимального управления летательных аппаратов, как атмосферных, так и космических, оказываются вырожденными ([39], гл. IX, X, стр. 188). Здесь аргументация более весомая, так как она подкрепляется ссылкой не на искусственные примеры, а на реальные, инженерные задачи. Однако подробный анализ этих примеров показывает, что их трактовка в [39] не бесспорна.

Задача об оптимальном маневре в вакууме. Имеется в виду маневр искусственного спутника, снабженного реактивным двигателем. Эта задача очень подробно исследована, ей посвящены целые монографии (например, [32], [48]), большое число конкретных задач (в них обычно требуется совершить тот или иной маневр с минимальным расходом топлива) исчерпывающе решено аналитически. Вырожденные решения появляются, например, в задаче об оптимальном переводе спутника с одной орбиты на другую. Оптимальное управление состоит в том, что двигатель включается только в определенных точках орбиты (в апогее и перигее), причем на возможно более короткий срок. Если задача идеализирована (нет ограничения сверху на секундный расход топлива и нет ограничения на время выполнения маневра), то оптимальное управление $u(t)$ (определяющее режим расхода топлива) вырождается в набор δ -функций, носители которых совпадают с моментами прохождения тела через апогей и перигей. Вообще, решение очень похоже на то, которое найдено (численно) в § 34 в связи с задачей о стабилизации спутника. Иногда это решение интерпретируют как пример скользящего режима. Это не очень удачная трактовка. Естественно связывать появление скользящих режимов с невыпуклостью вектограммы управляемой системы; в этой же задаче, как и в задаче § 34, функция Гамильтона H линейна по u , и, следовательно, вектограмма выпукла. Внешнее сходство полученного в этой задаче решения с характерной для скользящего режима картиной усиливается, если в качестве независимой переменной взять не время, а один из кеплеровых элементов орбиты. Эта переменная (удобная в аналитических исследованиях) при выключенном двигателе (т. е. при нулевом управлении) не меняется, поэтому при изображении управления как функции этого специального независимого аргумента, носители δ -функций оказываются примыкающими друг к другу, т. е. получается картина, аналогичная управлениям минимизирующей последовательности в задаче на стр. 83. По нашему мнению, не следует подобные примеры трактовать как скользящие режимы. И дело здесь не только в каких-то условностях. Дело в том, что настоящий скользящий режим связан с существенной для данной задачи невыпуклостью вектограммы системы. А в этом случае возникают и своеобразные трудности при численном решении (они поясняются на стр. 197). При численном решении задачи об оптимальном перелете, так же как и при численном решении задачи о стабилизации спутника (§ 34), в силу выпуклости вектограммы мы с этими трудностями не сталкиваемся.

В задаче о движении самолета ситуация совсем другая. Если читателя интересует "достаточно полный вид уравнений этой задачи, он может познакомиться с ними в § 30. Здесь достаточно будет пояснить, что производная одной компоненты x (угла наклона траектории) пропорциональна первой

степени управления u_2 (описывающего положение рулей), а производная другой (абсолютной величины скорости) пропорциональна квадрату управления (сопротивление воздуха). Таким образом, вектограмма системы — невыпуклая, и в принципе возможно появление скользящих режимов. Однако это еще не значит, что скользящий режим появится неизбежно. Во всяком случае в задачах, решенных в § 30 для системы с невыпуклой вектограммой, нет и намека на скользящий режим, и содержательный анализ этих задач показывает, что ожидать скользящий режим нет оснований. Этот анализ не так уж сложен и он подсказывает, как сконструировать для этой системы задачу, в которой почти наверняка появится скользящий режим. В самом деле, скользящий режим в системе типа

$$\dot{x}^1 = u; \quad \dot{x}^2 = -u^2; \quad |u| \leq 1$$

появится в том случае, когда по смыслу задачи, $u(t) \approx 0$, а $u^2(t)$ — максимально.

В частности, при движении реактивного самолета (§ 30) такая ситуация возникает при необходимости, выдерживая некоторый курс (в этом случае отклонение рулей в среднем равно нулю), произвести максимально возможное торможение (т. е. увеличить сопротивление воздуха за счет максимального отклонения рулей, все равно, положительного или отрицательного). Эта ситуация выглядит достаточно искусственно. В решавшихся автором задачах она не встречалась. В задачах, решавшихся другими для аналогичной системы уравнений, также не удалось найти такого примера. Речь, разумеется, идет о задачах, имеющих, так сказать, инженерное происхождение, а не придуманных специально для подтверждения той или иной точки зрения.

В [39] потенциальная возможность появления скользящего режима связана с несколько другой задачей. Например, можно взять уравнения задачи о вертикальном подъеме ракеты (§§ 28, 29), заменив во втором уравнении выражение для тяги Vu (u — секундный расход топлива) на $V(u)$, где $V(u)$ — нелинейная функция. При $V''(u) > 0$ возможен скользящий режим, в котором короткие отрезки времени с максимальным технически возможным расходом u перемежаются интервалами с $u=0$ (чем короче и чаще импульсы, тем ближе режим к оптимальному). При других формах $V(u)$ скользящих режимов нет. Остается не ясным, какие $V(u)$ соответствуют реальным двигателям, и реализованы ли технически какие-то аппроксимации скользящего режима, если он появляется в математической модели.

Вот этими соображениями и определяется отношение автора к скользящим режимам: вычислитель должен быть готов к встрече с этим объектом и должен иметь о нем достаточно ясное представление. В то же время класс практических задач (во всяком случае в настоящее время и в той мере, в какой он известен автору) еще не привел к настоятельной необходимости при реализации численных методов предусматривать и возможность скользящих режимов. Поэтому метод приближенного решения, применявшийся автором, на скользящие режимы не рассчитан. В § 23 будут в общих чертах описаны трудности нахождения скользящих режимов и возможные пути их преодоления. Хотя они приведут к определенным трудностям, но это, в сущности, трудности технического порядка, и в случае необходимости на них можно пойти.

§ 11. Вариационные задачи для ядерного реактора

Здесь мы рассмотрим вариационную задачу, отличающуюся от стандартной формой связи управления с фазовой траекторией. Подобные вариационные задачи возникают в связи с проектированием ядерных реакторов. Простейшая математическая модель

ядерного реактора приводит к системе уравнений

$$\begin{aligned} -\frac{d}{rdr} \frac{1}{D_1(u)} \frac{d\varphi_1}{dr} + A_{11}(u) \varphi_1 &= \lambda Q(u) \varphi_2, \\ -\frac{1}{rdr} \frac{1}{D_2(u)} \frac{d\varphi_2}{dt} + A_{21}(u) \varphi_1 + A_{22}(u) \varphi_2 &= 0, \quad 0 \leq r \leq R, \end{aligned} \quad (1)$$

с однородными краевыми условиями

при $r=0$:

$$d\varphi_1/dr = 0; \quad d\varphi_2/dr = 0, \quad (2)$$

при $r=R$:

$$\varphi_1 = 0; \quad \varphi_2 = 0.$$

Здесь $u(r)$ — управляющая функция, компоненты которой характеризуют относительную плотность веществ, заполняющих тело реактора. Потоки нейтронов $\varphi_1(r)$ (быстрые) и $\varphi_2(r)$ (медленные) определяются как компоненты первой *) собственной функции линейного дифференциального оператора (1)–(2); λ — соответствующее собственное число. Коэффициенты D_1 , D_2 , A_{11} , A_{12} , A_{21} , Q — заданные функции u . Переходя к стандартным обозначениям, запишем систему (1) в виде

$$\begin{aligned} -\frac{d(tx^1)}{tdt} + A_{11}[u(t)] x^3 &= \lambda Q[u(t)] x^4; \\ -\frac{d(tx^2)}{tdt} + A_{12}[u(t)] x^3 + A_{22}[u(t)] x^4 &= 0; \\ \frac{dx^3}{dt} - D_1[u(t)] x^1 &= 0; \\ \frac{dx^4}{dt} - D_2[u(t)] x^2 &= 0; \quad 0 \leq t \leq T. \end{aligned} \quad (1^*)$$

Краевые условия — линейные однородные

$$\Gamma x = 0: \quad x^1(0) = x^2(0) = 0; \quad x^3(T) = x^4(T) = 0. \quad (2^*)$$

В дальнейшем мы будем применять сокращенную запись

$$Lx = \lambda Qx, \quad x = \{x^1, x^2, x^3, x^4\}. \quad (3)$$

Решение конкретных задач для этой математической модели будет подробно описано ниже (см. § 38). Здесь мы ограничимся лишь вычислением функциональных производных для функционалов двух типов:

$$\text{I.} \quad F[u(\cdot)] \equiv \lambda. \quad (4)$$

$$\text{II.} \quad F[u(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt.$$

*) Соответствующей крайней левой точке спектра.

Заметим, что из (3) $x(t)$ определяется с точностью до нормировки. Обычно в формуле II $\Phi(x, u)$ не зависит от способа нормировки; ради общности мы этого предполагать не будем, нормировку считаем фиксированной тем или иным способом, норму v функции $x(t)$ будем считать элементом управления, после чего функционал типа II запишем в виде

$$\text{II.} \quad F[u(\cdot), v] \equiv \int_0^T \Phi[vx(t), u(t)] dt, \quad (4^*)$$

Дифференцирование функционалов в данной задаче осуществляется по стандартной схеме. Основными элементами этой техники являются:

1. Уравнение в вариациях, получаемое прямым варьированием уравнения (1*); его можно записать в форме

$$L\delta x - \lambda Q\delta x = \delta\lambda Qx + M\delta u, \quad \Gamma\delta x = 0. \quad (5)$$

Матрица $M[t]$ очевидным образом вычисляется на невозмущенной траектории $\{u(\cdot), x(\cdot)\}$; она линейно зависит от $x(t)$.

2. Тождество Лагранжа, справедливое для любой пары вектор-функций $\delta x(t)$, $\psi(t)$, удовлетворяющих условиям $\Gamma\delta x = 0$, $\Gamma^*\psi = 0$ (в дальнейшем мы будем рассматривать только такие пары):

$$\int_0^T \{\psi(L - \lambda Q)\delta x - \delta x(L^* - \lambda Q^*)\psi\} dt = 0. \quad (6)$$

Относительно оператора L в решавшихся нами задачах известно, что при любом имеющем физический смысл управлении $u(\cdot)$ крайняя точка спектра существует, вещественна, изолирована и однократна; кроме того, компоненты соответствующей собственной функции $\varphi_1(r)$, $\varphi_2(r)$ ($x^3(t)$, $x^4(t)$) не имеют перемен знака при $0 \leq r \leq R$ ($0 \leq t \leq T$).

3. Выражение для вариации функционала, полученное прямой вариацией определяющей его формулы

$$\begin{aligned} \delta F[\delta u(\cdot)] &\equiv \delta\lambda, \\ \delta F[\delta u(\cdot), \delta v] &\equiv \int_0^T \Phi_u[t] \delta u dt + \int_0^T \Phi_x[t] (v\delta x + \delta v x) dt. \end{aligned}$$

В последней формуле следует лишь преобразовать выражение $\int \Phi_x \delta x dt$ в интеграл только от вариации управления δv , $\delta u(t)$. В дальнейшем важную роль будет играть функция $\tilde{\psi}(t)$, являющаяся собственной функцией сопряженного оператора (соответствующей той же точке спектра λ):

$$L^*\tilde{\psi} - \lambda Q^*\tilde{\psi} = 0; \quad \Gamma^*\tilde{\psi} = 0. \quad (7)$$

Взяв в тождестве Лагранжа (6) $\tilde{\psi}$ и заменив в нем $L\delta x - \lambda Q\delta x$ правой частью (5), получим

$$\int_0^T \tilde{\psi} (\delta \lambda Qx + M\delta u) dt = 0,$$

откуда

$$\delta \lambda = - \frac{\int_0^T (\tilde{\psi}, M\delta u) dt}{\int_0^T (\tilde{\psi}, Qx) dt} = - \frac{\int_0^T (M^* \tilde{\psi}, \delta u) dt}{\int_0^T (\tilde{\psi}, Qx) dt}. \quad (8)$$

Этой формулой решается вопрос о дифференцировании функционала типа I. Заметим лишь, что в силу линейной зависимости M от x правая часть (8) не зависит от нормировки x . Теперь займемся преобразованием выражения типа

$$\int_0^T Y[t] \delta x(t) dt.$$

Пусть $\psi(t)$ — решение уравнения

$$L^* \psi - \lambda Q^* \psi = P^* Y[t]; \quad \Gamma^* \psi = 0, \quad (9)$$

где P^* — оператор проектирования, применяемый для того, чтобы вырожденное уравнение (9) имело решение. Результатом проектирования должна быть ортогональность

$$\int_0^T (P^* Y[t], x_1^*(t)) dt = 0.$$

В качестве конкретной реализации оператора P^* удобно взять преобразование

$$P^* Y \equiv Y[t] - \alpha [Y(\cdot)] Q^* \tilde{\psi}(t),$$

где

$$\alpha = \frac{\int_0^T (Y[t], x(t)) dt}{\int_0^T (\tilde{\psi}, Qx) dt}.$$

В этом случае уравнение (9) имеет решение, определенное, однакея с точностью до слагаемого, пропорционального $\tilde{\psi}(t)$. Избавимся от этой неопределенности, потребовав, чтобы ψ удовлетворяла ещ, условию

$$\int_0^T (\psi, Qx) dt = 0. \quad (10)$$

Заменяя в тождестве Лагранжа левые части уравнений (5) и (9), получим

$$\int_0^t (P^*Y, \delta x) dt = \int_0^t (\psi, \delta\lambda Qx + M\delta u) dt.$$

Теперь покажем, что

$$\int_0^t (P^*Y, \delta x) dt = \int_0^t (Y[t], \delta x(t)) dt.$$

Нужно иметь в виду, что $\delta x(t)$ определяется уравнением в вариациях через $\delta u(\cdot)$ (при $\delta\lambda$, однозначно определенном через $\delta u(\cdot)$ формулой (8)) не однозначно, но с точностью до слагаемого, пропорционального $x(t)$. Избавимся и здесь от неопределенности, наложив на δx условие

$$\int_0^t (\delta x, Q^*\tilde{\psi}) dt = 0,$$

что соответствует нормировке $x(\cdot)$ формулой

$$\int_0^t (x, Q^*\tilde{\psi}) dt = 1.$$

Теперь

$$\int_0^t (P^*Y, \delta x) dt = \int_0^t (Y - \alpha Q^*\tilde{\psi}, \delta x) dt = \int_0^t (Y, \delta x) dt.$$

Учитывая еще (10), получим окончательную формулу

$$\int_0^t (Y, \delta x) dt = \int_0^t (M^*\psi, \delta u) dt. \quad (11)$$

Заметим, что обычно функция Φ в выражении для функционала F типа II не зависит от нормировки x : при любом $\alpha \neq 0$

$$\Phi[x(t), u(t)] \equiv \Phi[\alpha x(t), u(t)].$$

Отсюда следует, что

$$\frac{\partial}{\partial \alpha} \Phi(\alpha x, u) = \Phi_x x = 0,$$

и дополнительное преобразование — проектирование правой части уравнения типа (9) — ненужно.

В некоторых решавшихся автором задачах отрезок $[0, R]$ разбивался на несколько частей точками $0 < R_1 < R_2 < R$;

на каждом интервале $(0, R_1)$, (R_1, R_2) , (R_2, R) использовались свои формулы для коэффициентов системы D_1, D_2, \dots , которые теперь следует обозначать $D_1(u, r), \dots$. Границы интервалов R_1, R_2, R не фиксированы и являются элементами управления; функционалы в этом случае следует обозначать $F[u(\cdot), R_1, R_2, R]$, и возникает необходимость находить производные $\partial F/\partial R_1, \partial F/\partial R_2, \partial F/\partial R$. Ниже мы проведем вычисления для более общего случая. В (1) будем считать r не независимым переменным, а одной из компонент фазового вектора, вводя дополнительную компоненту управления $v(t)$, и связав с ней $r(t)$ уравнением

$$dr/dt = v(t), \quad 0 \leq t \leq 1, \quad r(0) = 0; \quad v \geq v^-.$$

Оператор L теперь можно записать в виде

$$L(u, v, r)x = \lambda Q(u, v, r)x; \quad \Gamma x = 0.$$

Уравнение в вариациях имеет вид

$$L(u, v, r)\delta x + P\delta r - \lambda Q\delta x = \delta\lambda Qx + M\delta u + \mu\delta v.$$

Здесь $P[t]$ и $\mu[t]$ — матрицы $1 \rightarrow 4$, вычисление которых очевидным образом определяется прямым варьированием исходной системы уравнений. Вводя в дополнение к четырехмерной вектор-функции $\psi(t)$ еще скалярную функцию $\phi_0(t)$, запишем тождество Лагранжа:

$$\int_0^1 \left\{ \phi_0 \frac{d\delta r}{dt} + \psi(L - \lambda Q)\delta x \right\} dt = \int_0^1 \left\{ -\delta r \frac{d\phi_0}{dt} + \delta x(L - \lambda Q)^* \psi \right\} dt.$$

Здесь для δx и ψ приняты условия $\Gamma\delta x = 0$, $\Gamma^*\psi = 0$ и для δr и ϕ_0 — сопряженные условия $\delta r(0) = 0$, $\phi_0(1) = 0$. Заменяя $(L - \lambda Q)\delta x$ правой частью уравнения в вариациях, а $\delta\dot{r} = \delta v$, получим

$$\begin{aligned} \int_0^1 \{ \phi_0 \delta v + \psi [\delta\lambda Qx + M\delta u + \mu\delta v] \} dt = \\ = \int_0^1 \left\{ \delta r \left[-\frac{d\phi_0}{dt} - P\psi \right] + \delta x(L - \lambda Q)^* \psi \right\} dt. \end{aligned}$$

Используя эту формулу, можно получать необходимые выражения для производных, входящих в задачу функционалов. Например, для $F[u(\cdot), v(\cdot)] \equiv \lambda$ это приводит к таким вычислениям:

1. Определяется собственная функция $\tilde{\psi}$ оператора

$$(L^* - \lambda Q^*)\tilde{\psi} = 0; \quad \Gamma^*\tilde{\psi} = 0.$$

2. Находится функция $\phi_0(t)$ решением задачи Коши:

$$d\phi_0/dt + P\tilde{\psi} = 0; \quad \phi_0(1) = 0,$$

после чего имеем формулу

$$\delta\lambda = - \frac{\int_0^1 (\psi_0 + \mu\tilde{\psi}, \delta v(t)) dt + \int_0^1 (M^*\tilde{\psi}, \delta u) dt}{\int_0^1 (\tilde{\psi}, Qx) dt}$$

Аналогичные вычисления приводят и к производной функционала типа II. Эти формулы использовались автором в практических расчетах (см. [91], [92]).

§ 12. Задачи с уравнениями в частных производных

Рассмотрим задачу оптимального управления, в которой состояние объекта определяется решением уравнения с частными производными, а управление может входить в краевые (или начальные) данные, в правую часть или даже в выражения для коэффициентов уравнения. Различных вариантов постановок вариационных задач здесь великое множество *), и мы не будем стремиться все их проанализировать или сформулировать и разобрать максимально общую задачу. Мы ограничимся анализом одной частной задачи, да и в ней рассмотрим лишь технику вычисления функциональной производной, так как именно этим, в основном, отличаются друг от друга различные задачи.

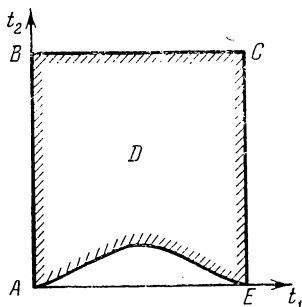


Рис. 10.

Итак, рассматривается система, состояние которой описывается функцией $x(t_1, t_2)$, определенной в двумерной области D (рис. 10) и удовлетворяющей в ней уравнению

$$\frac{\partial^2 x}{\partial t_1^2} + \frac{\partial^2 x}{\partial t_2^2} = f(t_1, t_2), \quad x|_{\Gamma} = 0. \quad (1)$$

Здесь Γ — граница D , сама же область D не фиксирована, ее форма и будет «управлением». Для простоты мы будем считать границы AB , BC и CE — фиксированными, а для участка AE примем уравнения

$$t_1 = \xi(s), \quad t_2 = \eta(s),$$

где

$$\frac{d^2 \xi}{ds^2} = u_1(s), \quad \frac{d^2 \eta}{ds^2} = u_2(s), \quad 0 \leq s \leq 1, \quad (2)$$

*) См. [15], [49].

с краевыми условиями, например, вида

$$\xi_1(0) = 0; \quad \xi_1(1) = 1; \quad \eta(0) = \eta(1) = 0. \quad (3)$$

Функции $u_1(s)$, $u_2(s)$ и будут «управлением» в этой задаче; что касается функционала, то мы ограничимся простой конструкцией

$$F[u(\cdot)] \equiv \int_0^1 \Phi \left[\frac{\partial x}{\partial t_2}(t_1, 1) \right] dt_1. \quad (4)$$

Целью дальнейшего является вычисление функций $w_1(s)$, $w_2(s)$, определяющих вариацию F в виде

$$\delta F[\delta u(\cdot)] = \int_0^1 [w_1(s) \delta u_1(s) + w_2(s) \delta u_2(s)] ds. \quad (5)$$

Алгоритм вычисления $w(s) = \{w_1, w_2\}$ будет получен последовательным анализом, состоящим из стандартных, в сущности, элементов:

1. Прямая вариация формулы (4) дает

$$\delta F[\delta u(\cdot)] = \int_0^1 Y[t_1] \frac{\partial \delta x(t_1, 1)}{\partial t_2} dt_1, \quad (6)$$

где $Y[t_1] = \Phi'[x_{t_2}(t_1, 1)]$ — функция, определенная на невозмущенном состоянии системы $x(t_1, t_2)$.

2. Уравнение в вариациях получается следующим образом. Пусть $x(t_1, t_2)$ — решение краевой задачи (1) в области $D[u(\cdot)]$, определяемой невозмущенным управлением $u(\cdot)$.

Пусть $\tilde{x}(t_1, t_2)$ — решение краевой задачи (1) в возмущенной области $\tilde{D} = D[u(\cdot) + \delta u(\cdot)]$. Функции x и \tilde{x} определены в разных областях, поэтому нельзя определить $\delta x = \tilde{x} - x$; для того чтобы корректно ввести вариацию δx , рассмотрим еще функцию $x^*(t_1, t_2)$, определенную в невозмущенной области D и отличающуюся от \tilde{x} на величину $O(\|\delta u\|^2)$ в тех точках, где существуют и x^* , и \tilde{x} , т. е. в $D^* = D \cap \tilde{D}$. Для $x^*(t_1, t_2)$ мы получим некоторую краевую задачу для уравнения Лапласа. Сейчас нам удобно будет описать возмущенную границу $\tilde{A}\tilde{E}$ скалярной функцией $\alpha(s)$, представляющей собой смещение $\tilde{A}\tilde{E}$ по нормали к AE в точке s . Другими словами, уравнение кривой $\tilde{A}\tilde{E}$ представим в виде

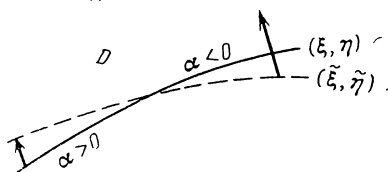
$$\begin{aligned} \xi(s) &= \xi(s) + \alpha(s) n_1(s), \\ \eta(s) &= \eta(s) + \alpha(s) n_2(s), \end{aligned} \quad (7)$$

где $\{n_1, n_2\}$ — внутренняя нормаль к AE . Разумеется, $\alpha(s)$ определяется через $\delta u(\cdot)$, и в дальнейшем $\alpha(s)$ будет исключена из формул. Заметим, что $|\alpha(s)| = O(\|\delta u\|)$.

Теперь определим $x^*(t_1, t_2)$ как решение уравнения $\Delta x^* = f$ в области D с теми же краевыми условиями, что и краевые условия для x на неизменных частях Γ , т. е. на AB, BC, CE . На границе (невозмущенной) AE для x^* поставим условие

$$x^*[\xi(s), \eta(s)] + \alpha(s) \partial x / \partial n = 0. \quad (8)$$

Покажем теперь, что на границе области $D \cap \tilde{D}$ $|x^* - \tilde{x}| = O(\|\delta u\|)$; следовательно, эта оценка будет справедлива во всей этой области. Прежде всего заметим, что $|x - x^*| = O(|\alpha|)$, так как на AE $x = 0$, а $x^* = O(|\alpha|)$. Поэтому $|\partial x / \partial n - \partial x^* / \partial n| = O(\|\delta u\|)$, и краевое условие с точностью до малых более высокого порядка, которые мы будем в дальнейшем опускать, можно записать в виде



$$x^* + \frac{\partial x^*}{\partial n} \alpha(s) = 0 \quad (8^*)$$

Рис. 11.

(можно было бы взять это условие вместо (8)).

В тех точках границы AE , где $\alpha(s) > 0$, т. е. возмущенная граница смещается внутрь D , значение функции x^* на кривой \tilde{AE} вычисляется по очевидной формуле

$$x^*|_{\tilde{AE}} = x^*_{AE} + \frac{\partial x^*}{\partial n} \alpha(s) + o(|\alpha|). \quad (9)$$

Таким образом, на этой части границы \tilde{AE} $x^* = o(|\alpha|)$. Рассмотрим те точки s , где $\alpha(s) < 0$, т. е. возмущенная граница выходит за пределы D (см. рис. 11, где возмущенная граница \tilde{AE} обозначена штриховой линией). Найдем значение \tilde{x} в точке невозмущенной границы $\{\xi(s), \eta(s)\}$ через значение ее в соответствующей точке возмущенной границы

$$\xi(s) = \xi(s) + \alpha(s) n_1(s), \quad \eta(s) = \eta(s) + \alpha(s) n_2(s).$$

В силу непрерывности нормаль \tilde{n} к \tilde{AE} в точке $\{\xi(s), \eta(s)\}$ с точностью до $O(|\alpha|)$ совпадает с нормалью $n(s)$ к AE в точке $\{\xi(s), \eta(s)\}$ (см. рис. 11) и, следовательно,

$$\tilde{x}(\xi, \eta) = \tilde{x}(\xi, \eta) - \alpha(s) \partial \tilde{x} / \partial n + o(|\alpha|). \quad (10)$$

В этой формуле можно $\partial \tilde{x} / \partial n$ заменить на $\partial x / \partial n$, после чего получим для \tilde{x} краевое условие на части AE , лежащей внутри D :

$$\tilde{x} + \alpha(s) \frac{\partial x}{\partial n} \Big|_{AE} = o(|\alpha|).$$

Рассмотрим область $D \cap \tilde{D}$; ее граница состоит из той части \tilde{AE} , где $\alpha(s) > 0$ и той части AE , где $\alpha(s) < 0$. На выделенной части \tilde{AE}

$\tilde{x}=0$, а $x^*=o(|\alpha|)$; на выделенной части AE $x^*+\alpha(s)\frac{\partial x}{\partial n}=0$ и $\tilde{x}+\alpha(s)\frac{\partial x}{\partial n}=o(|\alpha|)$. Отсюда заключаем, что всюду в $D \cap \tilde{D}$ $|x^*-\tilde{x}|=o(|\alpha|)$. Теперь в качестве вариации $\delta x(t_1, t_2)$ возьмем определенную всюду в D функцию

$$\delta x(t_1, t_2) \equiv x^*(t_1, t_2) - x(t_1, t_2), \quad \{t_1, t_2\} \in D. \quad (11)$$

Эта функция удовлетворяет в D уравнению

$$\partial^2 \delta x / \partial t_1^2 + \partial^2 \delta x / \partial t_2^2 = 0, \quad (12)$$

и краевым условиям

$$\begin{aligned} \delta x &= 0 \text{ на } AB, BC, CE, \\ \delta x + \frac{\partial x}{\partial n} \alpha(s) &= 0 \text{ на } AE \quad (\text{при } 0 \leq s \leq 1). \end{aligned} \quad (13)$$

Это и есть уравнение в вариациях, вывод которого был бы существенно проще, если бы форма области не менялась при вариации «управления».

3. Тождество Лагранжа

$$\iint_D (\Psi \Delta \delta x - \delta x \Delta \Psi) dt_1 dt_2 = \int_{\Gamma} \Psi \frac{\partial \delta x}{\partial n} dl - \int_{\Gamma} \delta x \frac{\partial \Psi}{\partial n} dl \quad (14)$$

позволит, как обычно, подобрав уравнение для Ψ , получить выражение, пока промежуточное, для $\delta F = \int_0^1 Y \frac{\partial \delta x}{\partial n} dl$ (dl — элемент длины дуги на границе Γ). В самом деле, возьмем в качестве $\Psi(t_1, t_2)$ решение краевой задачи

$$\left. \begin{aligned} \Delta \Psi &= 0 \text{ в } D, \\ \Psi &= 0 \text{ на } AB, CE, AE, \\ \Psi(t_1, 1) &= \Psi|_{BC} = Y[t_1]. \end{aligned} \right\} \quad (15)$$

Учитывая, что $\delta x|_{AB} = \delta x|_{BC} = \delta x|_{CE} = 0$; $\delta x|_{AE} = -\frac{\partial x}{\partial n} \alpha(s)$, получим из (14)

$$\int_0^1 Y[t_1] \frac{\partial \delta x}{\partial n} \Big|_{t_2=1} dt_1 + \int_{AE} \frac{\partial \Psi}{\partial n} \frac{\partial x}{\partial n} \alpha(s) dl = 0$$

и, таким образом,

$$\delta F = - \int_{AE} \frac{\partial \Psi}{\partial n} \frac{\partial x}{\partial n} \alpha(s) dl, \quad (16)$$

и для завершения вывода осталось в (16) выразить $\alpha(s)$ через $\delta u_1(s)$, $\delta u_2(s)$, а dl — через ds . Это уже чисто техническая выкладка.

Прежде всего вычислим нормаль к невозмущенной границе AE :

$$n(s) = \{n_1(s), n_2(s)\} = \frac{1}{\sqrt{\xi^2 + \eta^2}} \{\eta, -\xi\}.$$

Далее, $\alpha(s)$ вычисляется из условия пересечения прямой $\{\xi(s); \eta(s)\} + \alpha\{n_1(s); n_2(s)\}$ с возмущенной кривой \widetilde{AE} . Это дает два соотношения:

$$\begin{aligned}\xi(s + \delta s) + \delta\xi(s + \delta s) &= \xi(s) + \alpha(s)n_1(s), \\ \eta(s + \delta s) + \delta\eta(s + \delta s) &= \eta(s) + \alpha(s)n_2(s).\end{aligned}$$

Здесь учтено, что точке пересечения упомянутой выше прямой, проходящей через $\{\xi(s); \eta(s)\}$ перпендикулярно к AE , с \widetilde{AE} может соответствовать на \widetilde{AE} точка с возмущенным значением параметра $s + \delta s$. Полагая $\xi(s + \delta s) = \xi(s) + \xi\delta s$ и опуская малые второго порядка, получим систему для δs и $\alpha(s)$:

$$\begin{aligned}\xi\delta s + \delta\xi(s) &= \alpha n_1(s), \\ \eta\delta s + \delta\eta(s) &= \alpha n_2(s),\end{aligned}$$

откуда

$$\alpha(s) = \frac{1}{\sqrt{\eta^2 + \xi^2}} (\eta\delta\xi - \xi\delta\eta). \quad (17)$$

Теперь получим выражение для dl (оно очевидно):

$$dl = (d\xi^2 + d\eta^2)^{1/2} = (\xi^2 + \eta^2)^{1/2} ds.$$

Подставляя выражения для $\alpha(s)$ и dl в (16), получим формулу

$$\delta F = - \int_0^1 \frac{\partial \Psi}{\partial n} \frac{\partial x}{\partial n} (\eta\delta\xi - \xi\delta\eta) ds, \quad (18)$$

все еще не окончательную, так как нужно правую часть (18) преобразовать в интеграл от $\delta u(s)$.

Для $\delta\xi$, $\delta\eta$ имеем уравнения в вариациях

$$\begin{aligned}\frac{d^2\delta\xi}{ds^2} &= \delta u_1; & \delta\xi(0) &= \delta\xi(1) = 0, \\ \frac{d^2\delta\eta}{ds^2} &= \delta u_2; & \delta\eta(0) &= \delta\eta(1) = 0\end{aligned}$$

и тождества Лагранжа

$$\begin{aligned}\int_0^1 \left(\psi_1 \frac{d^2\delta\xi}{ds^2} - \delta\xi \frac{d^2\psi_1}{ds^2} \right) ds &= \psi_1 \frac{d\delta\xi}{ds} \Big|_0^1 - \delta\xi \frac{d\psi_1}{ds} \Big|_0^1, \\ \int_0^1 \left(\psi_2 \frac{d^2\delta\eta}{ds^2} - \delta\eta \frac{d^2\psi_2}{ds^2} \right) ds &= \psi_2 \frac{d\delta\eta}{ds} \Big|_0^1 - \delta\eta \frac{d\psi_2}{ds} \Big|_0^1.\end{aligned}$$

Взяв в качестве $\psi_1(s)$, $\psi_2(s)$ решения краевых задач

$$\left. \begin{aligned} \frac{d^2\psi_1}{ds^2} &= -\frac{\partial\Psi}{\partial n} \frac{\partial x}{\partial n} \eta(s); & \psi_1(0) &= \psi_1(1) = 0, \\ \frac{d^2\psi_2}{ds^2} &= \frac{\partial\Psi}{\partial n} \frac{\partial x}{\partial n} \xi(s); & \psi_2(0) &= \psi_2(1) = 0 \end{aligned} \right\} \quad (19)$$

и произведя стандартные исключения, получим окончательную формулу

$$\delta F = \int_0^1 \{ \psi_1(s) \delta u_1(s) + \psi_2(s) \delta u_2(s) \} ds. \quad (20)$$

Подведем итог, перечислив вычисления, которые необходимо произвести для нахождения производной функционала (4) по «форме границы».

1. Пусть имеется невозмущенное управление $\{u_1(s), u_2(s)\}$. Оно определяет область D , после чего может быть решена краевая задача (1) и найдены фазовая траектория $x(t_1, t_2)$ и значение функционала $F[u(\cdot)]$ по формуле (4).

2. Вычисляется функция $Y[t_1]$ (см. (6)).

3. Находится функция $\Psi(t_1, t_2)$ решением краевой задачи (15).

4. Находятся функции $\psi_1(s)$, $\psi_2(s)$ решением краевых задач (19) и

$$\frac{\partial F[u(\cdot)]}{\partial u_1(\cdot)} = \psi_1(s); \quad \frac{\partial F[u(\cdot)]}{\partial u_2(\cdot)} = \psi_2(s).$$

Г Л А В А П

МЕТОДЫ ПРИБЛИЖЕННОГО РЕШЕНИЯ ЗАДАЧ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

§ 13. Общие замечания к второй главе

Вторая глава книги (§§ 13—24) содержит описание характерных подходов к построению алгоритмов приближенного решения задач оптимального управления. Здесь перед автором стояли две противоречивые задачи: с одной стороны, дать читателю достаточно полное представление о возможных подходах к численному решению задач, с другой стороны, избежать чрезмерного многообразия, связанного, в частности, и с несущественными (и не всегда удачными) модификациями некоторых основных идей. Поэтому изложение некоторых методов часто дается для более общей задачи, чем это сделано в оригинальной работе. Однако такие обобщения делались автором, в основном, в тех случаях, когда это было связано лишь с чисто редакционными, не затрагивающими существа дела, коррективами. В то же время автор воздерживался от тех формально возможных обобщений класса задач (даже если они были отмечены в оригинальных работах), которые, по его мнению, приводят к существенному усложнению технической реализации метода и снижению его эффективности. Во всяком случае, автор старался предостеречь читателя от видимой простоты и легкости подобных обобщений, указывая на возникающие при этом осложнения. Эти легкость и простота существуют лишь до тех пор, пока речь идет о возможном «в принципе» решении задачи. Когда же дело доходит до реализации метода на ЭВМ, обнаруживаются значительные трудности (медленная сходимость, ненадежность результатов и т. д.). Так как многие идеи построения численных алгоритмов высказывались (в несущественно разных формах) разными авторами, возник вопрос о том, какой же публикации придерживаться. В этом случае автор обычно отдавал предпочтение тем, в которых дело было доведено до фактических расчетов. Это объясняется тем, что многие общие соображения носят настолько очевидный и элементарный характер, что вопрос о приоритете представляется неуместным и часто практически неразрешимым. Типичным примером подобных идей являются метод

штрафных функций, метод спуска по градиенту и некоторые другие. Видимо, лучшим решением вопроса о приоритете будет связать эти идеи с именами Ньютона, Эйлера и других классиков, или вообще ни с чьим именем не связывать. Разумеется, почти все алгоритмы, которые излагаются в этой главе, можно трактовать как конкретные реализации этих общих соображений. Однако реализация общей идеи неочевидна и неоднозначна и составляет, в сущности, основную часть работы по созданию эффективного численного метода.

Именно поэтому автор предпочитает работы, в которых общая идея доводится до вычислений, включая разработку соответствующей вычислительной технологии. В потоке работ, связанных с построением приближенных методов решения задач оптимального управления, можно выделить три главных направления. В книге этим направлениям уделено неравное внимание.

1. Первое направление связано с попытками так или иначе решить систему уравнений, образующих принцип максимума (подобно тому как точку минимума $f(x)$ можно искать, пытаясь прямо решать уравнение $f'(x)=0$). Этому направлению уделено немного места прежде всего потому, что надежных методов на этом пути пока создать не удалось. Причины неудач и возможные пути преодоления трудностей обсуждаются, однако все это еще нуждается в экспериментальной проверке.

2. Второе направление связано с построением минимизирующей последовательности траекторий, причем в качестве независимого аргумента берется не управление, а фазовая траектория (метод вариаций в фазовом пространстве). При таком подходе легко учитываются фазовые ограничения, однако возникают другие трудности. Этому направлению также уделено сравнительно небольшое место, так как имеются монографии [57], [86], посвященные, в основном, именно этому подходу.

3. Третье направление, имеющее, видимо, наибольшую литературу, связано с построением минимизирующей последовательности управлений. В книге оно отражено наиболее полно. Это связано как с естественностью выбора именно управления в качестве независимого аргумента, так и с тем, что разработанный и применявшийся в расчетах автором метод относится именно к этому направлению. Здесь есть свои сложности, особенно при решении задач с фазовыми ограничениями (с функционалами, не имеющими производных в смысле Фреше), однако читатель сможет убедиться, что они преодолимы, хотя дело это и не совсем простое.

Представим в общих чертах основные этапы развития численных методов решения задач оптимального управления, обратив особое внимание на то, как трудности реализации уже известных алгоритмов и растущие требования приложений определяют структуру новых методов. Начать историю численных методов в вариаци-

ционном исчислении нужно, видимо, с Эйлера. Именно он предложил заменить искомую функцию сеточной, а функционал — соответствующей разностной аппроксимацией. Правда, при этом преследовались теоретические цели, проведение необходимых для решения задач вычислений в то время было нереально. В дальнейшем этот метод был забыт, и в расчетах использовались методы Рунге, Галёркина и другие, аналогичные им. Они основаны на представлении искомого решения в виде сумм (с неопределенными коэффициентами) некоторого числа базисных функций. Умелый подбор базиса позволял обойтись двумя-тремя функциями и приводил к результату (в достаточно простых задачах) ценой не очень большого объема вычислений. Появление ЭВМ сняло, до известной степени, остроту вопроса о числе операций, и на первое место снова вышел метод конечных разностей Эйлера, благодаря его универсальности и слабой зависимости от аналитической формы задачи. В настоящее время большая часть приближенных методов оптимального управления так или иначе основана на идее Эйлера. Нужно, однако, понимать, что задача только и начинается после введения разностной аппроксимации. Основной вопрос приближенного решения в том, как найти минимум в полученной конечномерной задаче. Сама же идея конечно-разностной аппроксимации в наше время стала настолько тривиальной, что, даже реализуя ее на ЭВМ, не всегда понимают, что дело не сводится к нехитрому умению заменять производные разностями (см. в связи с этим §§ 25, 26, 36).

Первые методы приближенного решения задач оптимального управления были методами градиента в функциональном пространстве и применялись к простейшим задачам: найти

$$\min F_0[u(\cdot)] \text{ на траектории системы } \dot{x} = f(x, u), \quad x(0) = X_0. \quad (1)$$

В задаче (1) нет ни ограничений $u \in U$, ни условий $F_i = 0$, $i = 1, 2, \dots, m$. Вычисляется градиент $w_0(t)$ функционала F_0 , следующее управление есть $u(t) = sw_0(t)$, а шаг спуска s находится решением скалярной задачи $\min F_0[u(\cdot) - sw_0(\cdot)]$. Разумеется, задача (1) слишком проста; приложения приводят к более сложным, с условиями $u \in U$, $F_1 = \dots = F_m = 0$.

В принципе здесь нет никаких проблем, и метод «штрафных функций», предложенный впервые, видимо, Р. Курантом [38] именно в связи с решением вариационных задач еще в 1943 г., позволяет считать метод решения задачи (1) универсальным. Работа [38] породила мощный литературный поток, связанный с доказательством и обобщением теоремы о сходимости (при стремлении коэффициента штрафа к ∞), с различными формами штрафных функций (внешних, внутренних, комбинированных, использующих \ln , \exp и другие функции).

Формально метод штрафных функций решает все проблемы, однако при практической его реализации встретились серьезные трудности: медленная сходимость, ненадежность и грубость результатов. Причины этих неприятностей были поняты, и сторонники метода сосредоточили свои усилия на решении соответствующих вопросов вычислительной технологии: разработке надежных и эффективных методов поиска минимума для очень сложных, негладких, с «оврагами» и «хребтами» функций, методам подбора коэффициентов штрафа и тактике их изменения в процессе решения задачи. Эта работа продолжается, и в настоящее время ее перспективы еще не ясны. Идея метода штрафных функций имеет своих сторонников, которые надеются преодолеть технические сложности минимизации штрафного функционала. Одновременно начало развиваться и другое направление, в котором либо совсем не используют штрафных функций, либо стараются учесть методом штрафа как можно меньше условий. Разумеется, это потребовало определенного сужения класса задач. Легко были построены алгоритмы для задач, в которых имеется только ограничение $u(t) \in U$, а интегральных дополнительных условий (в частности, условий на $x(T)$) нет. В этом случае после вычисления градиента w_0 образуется семейство $u(s, t) = P_U[u(t) - Sw_0(t)]$, где P_U — оператор проектирования на U (в конечномерном пространстве). Далее S находится так же, как в простейшей задаче. Такие (или, в сущности, очень близкие) алгоритмы были предложены (под разными названиями) многими и применялись в расчетах (см., например, [43], [44]).

Что касается более общей задачи, то опять-таки можно сослаться на метод штрафных функций. Следующим шагом был метод проекции градиента в задачах с условиями $F_i[u(\cdot)] = 0$, $i = 1, 2, \dots, m$, причем все функционалы F_i — дифференцируемы по Фреше (см. § 18), а условий-неравенств $u \in U$ в задаче нет. Здесь дело уже осложняется тем, что нужно как-то выбирать шаг спуска s : увеличение s с понижением значения F_0 приводит к нарушению (в нелинейных задачах) условий $F_1 = \dots = F_m = 0$. Пионерами применения таких алгоритмов в прикладных задачах были, видимо, Т. М. Энеев в СССР, Брайсон, Кэлли в США (конец 50-х годов). Что касается условий $u \in U$, то приходилось использовать либо опять штрафные функции, либо другие приемы (замена управления, преобразование Валентайна), имеющие свои отрицательные стороны (см. § 18) и не удовлетворяющие достаточно требовательных вычислителей. Работа продолжалась. В частности, Брайсон, Дэнхем и Дрейфус [13] для задач с условиями $\{G[x(t), u(t)] \leq 0 \text{ при всех } t\}$ ввели проектирование градиента на линейное многообразие $\{G_x \delta_x(t) + G_u \delta u(t) = 0 \text{ при } t \in [t_a, t_w]\}$, причем t_a, t_w — искомые (вместе с $u(\cdot)$) неизвестные. В последние годы в работах группы Miele ([53] содержит их обзор и библиографию) исполь-

зуется, в сущности, то же самое, только вместо явного введения в задачу гипотезы о структуре решения ($G(x, u) = 0$ при $t \in [t_\alpha, t_\omega]$, $G(x, u) < 0$ вне $[t_\alpha, t_\omega]$) с искомыми t_α, t_ω используется преобразование Валентайна. Едва ли это можно считать прогрессом. Во всяком случае, в [13] (1964 г.) метод испытывался на реальной прикладной задаче, а в [53] испытания проводятся пока на простых модельных задачах.

Дальнейшее развитие численных методов было связано со стремлением учесть как ограничения $u \in U$, так и дополнительные условия $F_1 = \dots = F_m = 0$ (обычно они имели форму условий на правом конце траектории $\Phi; [x(T)] = 0$). Кроме того, предметом особых усилий были ограничения в фазовом пространстве ($\Phi[x(t)] \leq 0$ при всех t) и ограничения общего вида ($\Phi[x(t), u(t)] \leq 0$). Именно связанные с учетом таких условий трудности стимулировали развитие методов вариаций в фазовом пространстве (§§ 15, 16; см. также [55], [56]). Эти методы настолько успешно справлялись с ограничениями в фазовом пространстве, что возникающие на этом пути серьезные трудности (отсутствие сходимости в методе локальных вариаций, медленная сходимость, ненадежные и неточные результаты, учет условий $u \in U$ в какой-то мере выпали из поля зрения. К тому же на основании спорных оценок числа операций был сделан вывод о преимуществе метода локальных вариаций перед другими итерационными методами (метод трубки, см. § 16), и эта форма вариаций в фазовом пространстве стала, видимо, основным вычислительным инструментом.

С этими же трудностями связано возвращение к методу Эйлера в его самой бесхитростной форме. Имеется в виду то направление, которое получило название «метод математического программирования в теории оптимального управления». Почти всякий метод приближенного решения задач оптимального управления может быть охвачен этим термином, поэтому следует уточнить, о чем идет речь. Это — направление, в котором задачу заменяют конечно-разностной, переписывают все ограничения задачи в виде ограничений на значения сеточных функций, интегралы заменяют суммами, и, получив конечномерную задачу минимизации при наличии ограничений, ссылаются на возможность ее решения хорошо разработанными методами математического программирования. Последние представляют тему огромного числа статей и монографий, но это как раз и свидетельствует о том, что надежных методов решения общей задачи минимизации нет.

Автор начал приближенное решение задач оптимального управления в 1962 г., когда на очереди было решение задач в достаточно общей постановке: с условиями $u \in U$, $F_1 = \dots = F_m[u(\cdot)] = 0$, с функционалами, не имеющими производных Фреше.

Здесь будет разъяснена идеологическая основа разработанного автором метода. Следующие положения лежат в его основе.

1. С точки зрения вычислительной математики трудность задачи определяется не ее формой, а дифференциальными свойствами входящих в задачу функций. Поэтому не следует употреблять приемов, упрощающих внешнюю форму задачи ценой ухудшения свойств гладкости функций (штрафные функции, преобразование Валентайна и т. п.). Разумеется, это приводит к употреблению более сложных (формально) алгоритмов.

2. Вычислительные методы так или иначе связаны с аппроксимацией функциональных пространств конечномерными. Эффективность метода существенно зависит от того, как используется конкретная функциональная природа того или иного объекта. В задаче оптимального управления объединены объекты с разными функциональными свойствами: дифференцируемая функция $x(t)$, измеримая $u(t)$, дифференциальные связи, интегральные связи, функционалы, дифференцируемые по Фреше и дифференцируемые лишь по направлениям; среди последних есть функционалы типа $\max \Phi[x(t)]$, а есть существенно другие: $\max \Phi[x(t), u(t)]$. Каждый из объектов требует своего подхода. На разностном уровне различия между этими объектами, на первый взгляд, стираются, и есть возможность все их трактовать единым образом. Именно эта точка зрения лежит в основе методов математического программирования в оптимальном управлении (см., например [75]). Однако при реализации таких единообразных подходов в достаточно сложных задачах она приводит к серьезным трудностям (см. в связи с этим §§ 25, 34, 36).

3. Собственно вычислительный аппарат алгоритма должен быть адекватен задаче. Мы имеем дело с неклассической задачей, в условия которой входят неравенства. Поэтому привычный вычислительный аппарат линейной алгебры, ориентированный на решение задач в терминах равенств, недостаточен, следует привлечь аппарат линейного программирования. Этим работа автора существенно отличается от основной массы алгоритмов, которые так или иначе связаны с привычным аппаратом линейной алгебры.

Применимость его в неклассических задачах обеспечивается за счет штрафных функций, преобразования Валентайна и других приемов того же сорта (см., в частности, § 39). Линейное программирование есть вычислительный аппарат для задач с неравенствами, а не метод решения только экономических задач.

4. При решении задач оптимального управления возникают специфические задачи линейного программирования. Надо быть готовым к тому, что стандартные методы решения таких задач окажутся недостаточно эффективными и придется разрабатывать специализированные.

Как эти положения реализуются в конкретной вычислительной работе, читатель узнает из содержания §§ 19—21 и третьей главы.

§ 14. Методы решения краевой задачи для π -системы

Рассмотрим следующую вариационную задачу: на траектории управляемой системы

$$\begin{aligned} \frac{dx}{dt} &= f(x, u), \quad 0 \leq t \leq T, \\ x(0) &= X_0; \quad u(t) \in U \end{aligned} \quad (1)$$

минимизировать функционал

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (2)$$

при дополнительных условиях:

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m. \quad (3)$$

Все функционалы F_i будем считать дифференцируемыми, для определенности ограничимся конструкциями вида

$$F_i[u(\cdot)] \equiv \int_0^T \Phi^i[x(t), u(t)] dt, \quad (4)$$

представляя, если нужно, в этой же форме и функционал

$$F[u(\cdot)] \equiv \Phi[x(t')] = \int_0^T \Phi[x(t)] \delta(t - t') dt.$$

В этом случае принцип максимума утверждает существование функции $\psi(t)$, являющейся решением задачи:

$$\frac{d\psi}{dt} + f_x[t]\psi = - \sum_{i=0}^m g_i \Phi_x^i[t], \quad \psi(T) = 0, \quad (5)$$

определенной с точностью до параметров g_1, g_2, \dots, g_m ($g_0 = -1$), причем оптимальное управление $u(t)$ удовлетворяет условию

$$H[x(t), \psi(t), u(t)] = \max_{u \in U} H[x(t), \psi(t), u], \quad (6)$$

где

$$H[x, \psi, u] \equiv \sum_{i=0}^m g_i \Phi^i[x, u] + (f(x, u), \psi). \quad (7)$$

Определение. Систему уравнений

I. $\dot{x} = f(x, u),$

II. $-\dot{\psi} = f_x^*(x, u)\psi + \sum_{i=0}^m g_i \Phi_x^i(x, u), \quad (8)$

III. $H[x(t), \psi(t), u(t)] = \max_{u \in U} H[x(t), \psi(t), u]$

называют Π -системой.

Формально Π -система замыкается конечными соотношениями

$$\begin{aligned} 1. \quad & x(0) = X_0 \quad (\Gamma(x) = 0), \\ 2. \quad & \psi(T) = 0 \quad (\Gamma_x^* \psi = 0), \\ 3. \quad & F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m. \end{aligned} \quad (9)$$

Под формальным замыканием имеется в виду простое сравнение «степеней свободы» для системы (8) с числом конечных соотношений (9). Пусть из (8; III) можно однозначно определить $u(t)$ как функцию $x(t)$, $\psi(t)$ и g . Уравнение $H[x, \psi, u^*] = \max_{u \in U} H[x, \psi, u]$ разрешается относительно $u^*: u^* = V(x, \psi, g)$. Тогда система (8; I), (8; II) превращается (формально) в систему $2n$ уравнений для n -мерных вектор-функций $x(t)$ и $\psi(t)$:

$$\begin{aligned} \dot{x} &= f[x, V(x, \psi, g)], \\ \dot{\psi} + f_x[x, V(x, \psi, g)]\psi &= - \sum_{i=0}^m g_i \Phi^i[x, V(x, \psi, g)], \end{aligned}$$

множество решений которой определяется заданием, например, $x(0)$, $\psi(0)$ и g_1, \dots, g_m , т. е. имеет размерность $2n+m$. Наличие $2n+m$ конечных соотношений (9) формально делает выбор $x(0)$, $\psi(0)$, g однозначным и, тем самым, однозначно определяет $x(t)$, $u(t)$ — искомую оптимальную траекторию.

Таким образом, решение вариационной задачи формально сведено к решению краевой задачи для Π -системы (8) — (9). Хотя эта формальная схема рассуждений содержит ряд нестрогих заключений (на них мы еще обратим внимание), не вызывает никаких сомнений, что разработка надежных методов решения Π -систем была бы существенным вкладом в численное решение вариационных задач. К сожалению, здесь встретились значительные трудности, преодолеть которые пока не удалось. Однако следует сразу же отметить, что наиболее точные и аккуратные численные решения вариационных задач связаны именно с решением соответствующих Π -систем; правда, удалось это, несмотря на многочисленные попытки, в очень редких случаях.

Если иметь в виду достаточно общий случай, то естественным подходом к решению краевой задачи является, видимо, следующий.

Введем вектор искомых параметров $\xi = \{\psi_1(0), \dots, \psi_n(0), g_1, \dots, g_m\}$, задание которого формально дополняет Π -систему (8) до задачи Коши. Пусть при любом заданном ξ эта задача Коши интегрируется (численно) и однозначно определяет $x(t)$, $\psi(t)$, $u(t)$ и, следовательно, $\psi(T)$ и $F_i[u(\cdot)]$. Таким образом, этой процедурой численного интегрирования устанавливается функцио-

нальная зависимость $(n+m)$ -мерного вектора $z = \{\psi_1(T), \dots, \psi_n(T), F_1[u(\cdot)], \dots, F_m[u(\cdot)]\}$ от $(m+n)$ -мерного вектора ξ :

$$z = Z(\xi),$$

и теперь формально решение краевой задачи для Π -системы сводится к решению системы нелинейных уравнений $Z(\xi) = 0$ с достаточно сложным определением функциональной зависимости $Z(\xi)$.

Заметим, что в случае, когда в постановке вариационной задачи не все значения $x(0)$ фиксированы, свободные компоненты $x(0)$ включаются в вектор параметров ξ , увеличивая его размерность, а вектор «условий» Z расширяется добавлением соответствующих условий трансверсальности.

Наиболее надежным общим методом решения систем нелинейных уравнений является метод Ньютона: имея некоторое приближение ξ^0 , ищем поправку $\delta\xi$ так, чтобы

$$Z(\xi^0 + \delta\xi) \simeq Z(\xi^0) + Z_\xi(\xi^0) \delta\xi = 0,$$

т. е. $\delta\xi = -Z_\xi^{-1}Z(\xi^0)$, и следующее приближение ξ^1 есть

$$\xi^1 = \xi^0 - Z_\xi^{-1}Z(\xi^0).$$

Рассмотрим трудности, возникающие при фактической реализации этой схемы и возможные пути их преодоления; хотя эти приемы и не позволяют решить задачу в общем случае, они оказываются полезными при решении частных, сравнительно простых задач.

1. В ы ч и с л е н и е м а т р и ц ы $Z_\xi : (n+m) \rightarrow (n+m)$.

В общем случае единственным способом решения задачи вычисления Z_ξ , учитывая «неявный» способ задания функции $Z(\xi)$, является численное дифференцирование. Таким образом, вычисление $Z(\xi)$ и Z_ξ требует, по меньшей мере, $(n+m+1)$ -кратного интегрирования задачи Коши. Учитывая возможности современных ЭВМ, следует признать это обстоятельство отнюдь не самым неприятным по сравнению с остальными.

2. С х о д и м о с т ь м е т о д а Н ь ю т о н а л и ш ь в о к р е с т н о с т и р е ш е н и я. Эти вопросы разбираются в § 43. Описанная там модификация метода Ньютона

$$\xi^1 = \xi^0 - sZ_\xi^{-1}Z(\xi^0)$$

с выбором параметра s как решения одномерной задачи

$$\min_s \|Z(\xi^0 - sZ_\xi^{-1}Z)\|$$

при подходящем определении $\|Z\|$ позволяет повысить надежность метода Ньютона и расширить область сходимости.

3. О т с у т с т в и е т е о р е м ы о е д и н с т в е н н о с т и. Для того чтобы существовала зависимость $Z(\xi)$, по меньшей мере

необходима единственность решения задачи Коши для Π -системы (если не для всех значений начальных данных ξ , то хотя бы в окрестности искомого решения). Однако в общем случае такая теорема не доказана. Более того, она и не может быть доказана, так как в довольно простых задачах единственности нет, и это отсутствие единственности существенно, так как искомая оптимальная траектория часто как раз и входит в бесконечное множество траекторий, определяемое некоторыми начальными данными ξ^0 , для которых нарушается единственность. Приведенное выше утверждение основано на примере, подробно рассмотренном в § 28; есть все основания считать этот пример достаточно типичным.

Нарушение единственности решения задачи Коши связано с тем, что определяемая уравнением принципа максимума (6) зависимость $V(x, \phi, g)$ при некоторых значениях аргументов не удовлетворяет условию Липшица (с показателем 1), обеспечивающему применение стандартной теоремы единственности. Типичным является, например, наличие разрывов в $V(x, \phi, g)$ при особых значениях аргументов. И хотя «почти для всех» x, ϕ, g зависимость $V(x, \phi, g)$ непрерывна и дифференцируема, упомянутых разрывов часто оказывается достаточно для того чтобы лишить описанную выше формальную процедуру решения краевой задачи для Π -системы всяких шансов на успех.

Есть два способа бороться с этой неприятностью. Первый способ — строго выпуклая аппроксимация — состоит в замене исходной системы уравнений близкой к ней системой

$$dx/dt = f(x, \tilde{u}, \varepsilon), \quad \tilde{u} \in U(\varepsilon), \quad (1^*)$$

где \tilde{u} — некоторое новое управление, быть может большей размерности, чем исходное u , а $f(x, \tilde{u}, \varepsilon)$ — новые правые части. Разумеется, замена исходной системы новой может быть оправдана лишь при условии достаточной малости вводимых этим искажений. Предполагается, что для всех $x, \tilde{u} \in U(\varepsilon)$ найдется $u \in U$ такое, что

$$\|f(x, u) - f(x, \tilde{u}, \varepsilon)\| \leq \varepsilon,$$

и наоборот, любое $f(x, u)$ может быть с точностью до $O(\varepsilon)$ заменено на $f(x, \tilde{u}, \varepsilon)$. Смысл эта замена имеет лишь в том случае, если область

$$f(x, U(\varepsilon), \varepsilon) = \{f(x, \tilde{u}, \varepsilon)\}_{\tilde{u} \in U(\varepsilon)}$$

является строго выпуклой, имеющей гладкую границу с ограниченным (равномерно по x) радиусом кривизны. В этом случае при любом ϕ

$$\max_{\tilde{u} \in U(\varepsilon)} (\phi, f(x, \tilde{u}, \varepsilon))$$

достигается в единственной точке $\tilde{u}(x, \phi)$, а ограниченность радиуса кривизны границы $f(x, U(\varepsilon), \varepsilon)$ обеспечивает выполнение

условия Липшица для функции $\tilde{a}(x, \psi)$. Формально единственность задачи Коши для новой, аппроксимирующей П-системы восстановлена, и появляется надежда все-таки решить и краевую задачу.

Почти во всех прикладных задачах, известных автору, области $f(x, U)$ — не строго выпуклы. Одной из причин этого является то, что, как правило, r (размерность u) меньше n (размерности x и f), а гладкое отображение простой ограниченной области U r -мерного пространства в n -мерное есть r -мерное многообразие и, следовательно, не может содержать n -мерной сферы (что есть необходимое условие строгой выпуклости множества в n -мерном пространстве).

Что касается фактического осуществления аппроксимации (1*), то, не разрабатывая общих приемов, заметим, что в прикладных задачах она обычно осуществляется без особого труда. Ограничимся простыми примерами, поясняющими суть дела.

Пример 1. В двумерном пространстве квадрат $(|u_1| \leq 1) \times (|u_2| \leq 1)$, не являющийся строго выпуклым множеством, может быть сколь угодно точно аппроксимирован строго выпуклым овалом $u_1^{2k} + u_2^{2k} \leq 1$, $k \gg 1$ ($\varepsilon \sim 1/k$).

Ошибка аппроксимации может быть оценена на диагонали $u_1 = u_2$: овал пересекает диагональ в точке с координатами $u_1 = u_2 = (0, 5)^{2k}$ и $\varepsilon \sim 1 - (0, 5)^{2k}$.

Пример 2. Отрезок $(|u_1| \leq 1) \times (u_2 = 0) \times (u_3 = 0)$ в трехмерном пространстве аппроксимируется эллипсоидом

$$u_1^2 + A^2(u_2^2 + u_3^2) \leq 1, \quad A \gg 1.$$

Однако это чисто формальное преодоление неединственности с вычислительной точки зрения не следует оценивать слишком высоко. Дело в том, что радиус кривизны некоторых частей границы области $f(x, U, \varepsilon)$ оказывается очень большим, стремящимся к ∞ при $\varepsilon \rightarrow 0$. Хотя $\tilde{a}(x, \psi)$ и удовлетворяет условию Липшица

$$\|\tilde{a}(x', \psi') - \tilde{a}(x, \psi)\| \leq C_\varepsilon \{\|x' - x\| + \|\psi' - \psi\|\},$$

но с очень большой постоянной $C_\varepsilon \rightarrow \infty$ при $\varepsilon \rightarrow 0$. Это приводит к чрезвычайно «капризному» характеру зависимости $Z(\xi)$: малые изменения ξ (данных Коши П-системы) приводят к огромным, нерегулярным изменениям правого конца траектории — точки $Z(\xi)$. Поэтому следует ожидать значительных затруднений в сходимости метода Ньютона.

Хотя аппроксимация (1*) и применялась в практических расчетах (мы разберем соответствующие примеры в § 27), автор не знает случаев, когда это делалось при существенной неединственности задачи Коши и помогло преодолеть возникающие вычислительные трудности.

Вообще следует иметь в виду, что в вычислительной математике нет таких ситуаций, чтобы трудности, отсутствовавшие в аппроксимирующей ϵ -задаче, внезапно появлялись в предельной, соответствующей $\epsilon=0$. И в данном случае разделение задач на два типа — строго выпуклые, в которых при сколь угодно малом ϵ единственность задачи Коши обеспечена, и предельные, не являющиеся строго выпуклыми, в которых нет единственности, нельзя, безусловно, трактовать, как возможность использовать метод Ньютона в аппроксимирующей Π -задаче. Дело в том, что следует принять во внимание очень важный фактор — эффективность вычислительного алгоритма. К сожалению, мы не имеем здесь эффективных оценок, однако, ясно, что стремление $\epsilon \rightarrow 0$ сопровождается ростом вычислительных трудностей.

Метод аппроксимации (1*) привлекает внешней простотой преодоления трудностей, но эта простота обманчива и обычно сменяется сложностью, когда алгоритм реализуется на ЭВМ.

Другой путь борьбы с неединственностью носит более принципиальный характер и, если его удастся реализовать, приводит к хорошим практическим результатам. Однако его реализация весьма трудна, требует индивидуального анализа решаемой задачи. Общих рецептов здесь нет. Поэтому мы ограничимся лишь кратким изложением существа дела. Метод состоит в качественном описании множества решений Π -системы, которое часто допускает однозначную параметризацию, причем число параметров равно числу неиспользованных конечных соотношений в краевой задаче для Π -системы. Формально это совпадает с приведенной выше и отвергнутой схемой рассуждений. Но дело в том, что начальные данные задачи Коши не могут быть взяты в качестве этой системы параметров. Нужно искать другие, успех здесь требует тщательного качественного анализа задачи.

Ограничимся этими общими замечаниями; более точное представление о том, что имеется в виду, дают примеры, где эту программу удалось реализовать и получить весьма точные решения сложных вариационных задач. Стоит отметить, что качественный анализ задачи, позволяющий построить необходимую параметризацию семейства решений Π -системы, может быть основан как на аналитической работе, так и на изучении приближенных решений задачи, полученных каким-либо численным методом, эффективным, хотя и дающим относительно грубое решение.

4. Неоднзначность отображения $Z(\xi)$. Нет никаких оснований считать отображение $Z(\xi)$ взаимно однозначным всюду, даже если единственность задачи Коши гарантирована. В самых простых ситуациях, как показывают примеры, возможны различные типы вырождения, например, $(n+m)$ -мерная сфера в пространстве ξ (напомним, что $(n+m)$ — размерность ξ) может отображаться в $(n+m-1)$ -мерное многообразие в простран-

стве z , а это приводит к тому, что метод Ньютона с описанными выше модификациями перестает работать: как говорят вычислители, метод застревает (формально это сказывается в несуществовании, например, Z_{ξ}^{-1}). Пример подобной ситуации подробно рассмотрен в § 27.

5. Выбор начального приближения. Модификация метода Ньютона не снимает проблемы подбора достаточно хорошего начального приближения, хотя и заметно ослабляет остроту этого вопроса. Опыт показал, что использование каких-либо содержательных соображений в целях нахождения хорошего начального приближения ξ^0 крайне затруднительно даже в тех задачах, где подбор разумного приближения в терминах управляющей функции $u(t)$ сравнительно прост. Пожалуй, единственным выходом является решение задачи каким-либо иным методом, достаточно надежно дающим относительно грубое приближенное решение. Такие приближенные методы в настоящее время разработаны, отличительной их чертой является то, что они дают хорошее приближение к искомому решению с точки зрения фазовой траектории $x(t)$ и значений функционалов задачи $F, [u(\cdot)]$, однако обычно довольно грубое с точки зрения управляющей функции $u(t)$. Фигурирующий в принципе максимума вектор g тоже, как правило, получается с хорошей точностью. Создание приближенного метода решения задач оптимального управления, соединяющего надежность и эффективность с хорошей точностью по всем компонентам задачи возможно, видимо, лишь комбинированием методов грубого поиска минимума с последующим уточнением точного вида решения, основанным на использовании характеризующих его уравнений типа принципа максимума или уравнения Эйлера.

§ 15. Метод вариаций в фазовом пространстве

Н. Н. Моисеевым и его сотрудниками был разработан метод приближенного решения вариационных задач, являющийся, по существу, методом спуска в фазовом пространстве. В настоящем параграфе будет описана принципиальная схема метода в его наиболее общей форме, хотя практические расчеты велись не по этой схеме, а по некоторым ее упрощенным модификациям; им будет посвящен следующий параграф. Дело в том, что полный и теоретически обоснованный алгоритм Н. Н. Моисеева практически нереализуем для прикладных задач на современных ЭВМ, однако содержащиеся в нем идеи породили упоминавшиеся выше упрощенные модификации. Последние уже реализуемы и применялись на практике, но вопросы их обоснования встречают серьезные препятствия по существу дела.

Общая схема метода. Решается следующая задача оптимального управления: минимизировать аддитивный функционал

$$F_0[u(\cdot), x(\cdot)] \equiv \int_0^T f^0[x(t), u(t)] dt \quad (1)$$

на траекториях управляемой системы

$$\begin{aligned} \dot{x} &= f(x, u), \quad 0 \leq t \leq T, \\ x &= \{x^1, \dots, x^n\}, \quad f = \{f^1, \dots, f^n\} \end{aligned} \quad (2)$$

при краевых условиях

$$x(0) = X_0, \quad x(T) = X_1, \quad (3)$$

учитывая как геометрическое ограничение управления

$$u(t) \in U \text{ при всех } t, \quad (4)$$

так и ограничение в фазовом пространстве

$$x(t) \in G \text{ при всех } t. \quad (5)$$

В (4) и (5) можно писать $U(t)$, $G(t)$; так как с этим обобщением никаких серьезных осложнений не связано (во всяком случае, при гладких зависимостях U и G от t), то мы не станем усложнять изложения подобной общностью. U и G считаются ограниченными замкнутыми областями.

На интервале $[0, T]$ вводится счетная сетка, для простоты равномерная: $t_0 = 0$, $t_1, \dots, t_N = T$; $t_i = i\tau$; $\tau = T/N$. В каждой точке t_i определяется экземпляр сетки в фазовом пространстве, покрывающий область G с некоторой густотой, определяемой шагом h в фазовом пространстве; совокупность точек i -й сетки $\{x_j^i\}$ будем обозначать S^i . Заметим, что индекс j , который можно считать, например, n -мерным мультииндексом, принимает по числу узлов сетки S^i $O\left(\frac{1}{h^n}\right)$ разных значений.

Предположим еще, что $X_0 \in S^0$, $X_1 \in S^N$.

Элементарная операция. Предполагается, что для любой пары x_j^i, x_k^{i+1} может быть решена задача того же типа, (1)—(2)—(3)—(4)—(5), однако на малом интервале $[t_i, t_{i+1}]$ и с левым и правым концами траектории в точках x_j^i, x_k^{i+1} соответственно. Задачу не обязательно решать очень точно, что, вместе с малостью интервала $[t_i, t_{i+1}]$, позволяет во многих случаях без особого труда получить число $\delta F_{j,k}^{i+1/2}$ — цену в терминах функционала F_0 оптимального перехода из $\{t_i, x_j^i\}$ в $\{t_{i+1}, x_k^{i+1}\}$ *). Имея набор этих

*) Вычисление $\delta F_{j,k}^{i+1/2}$ и называется элементарной операцией.

чисел $\delta F_{j,k}^{i+1/2}$, можно строить некоторую приближенно оптимальную траекторию исходной задачи (2)—(5). Заметим, однако, что объем этой информации очень велик: чисел $\delta F_{i,k}^{i+1/2}$ примерно $O\left(N \frac{1}{h^{2n}}\right)$, в общем случае столько раз и следует проделать вычисление $\delta F^{i+1/2}$, т. е. использовать элементарную операцию (в действительности, много меньше, так как для большинства пар $x_{j_i}^i, x_k^{i+1}$ никакой траектории не существует).

Выбор (τ, h) -оптимальной траектории. Ставится следующая задача: найти последовательность точек—узлов сеток S^i :

$$x_{j_0}^0 = X_0, \quad x_{j_1}^1, x_{j_2}^2, \dots, x_{j_i}^i, \dots, x_{j_N}^N = X_1 \quad (x_{j_k}^k \in S^k), \quad (6)$$

таким образом, чтобы обеспечить минимальность

$$F_0 \equiv \sum_{i=0}^{N-1} \delta F_{j_i, j_{i+1}}^{i+1/2}.$$

Решение этой задачи в принципе не так уж сложно — алгоритм дискретного динамического программирования, подробно описанный в § 44, приводит к цели с затратой числа операций в общем случае порядка $O(Nh^{-2n})$. Последовательность точек (6) и объявляется оптимальной траекторией задачи (1)—(5); разумеется, речь идет о приближенно оптимальной траектории, точность зависит от шагов сетки τ и h . Если элементарная операция реализована точным решением задачи типа (1)—(5) на малом интервале $[t_i, t_{i+1}]$, то мы имеем дело с точной траекторией управляемой системы (2), проходящей через узлы $x_{j_i}^i$ в моменты t_i ; обычно элементарная операция реализуется не абсолютно точно, и узлы (6), соединенные, например, отрезками прямых, представляют некоторую аппроксимацию решения системы $\dot{x}=f$. Если нас интересует не только оптимальная траектория (6), но и реализующее ее управление $u(t)$, то его можно восстановить по узлам (6) с помощью той же «элементарной» операции. Следует прежде всего подчеркнуть ту легкость, с которой данный метод справляется со всеми ограничениями на фазовую часть траектории, будь то ограничения на правом конце траектории ($x(T)=X_1$) или еще более сложные ограничения типа $x(t) \in G$ при всех t . В известной монографии [57] отражена история развития методов приближенного решения задач оптимального управления группой ВЦ АН СССР под руководством Н. Н. Моисеева. Работа начиналась с естественной попытки строить минимизирующие последовательности управляющих функций. После первых успехов в решении простейших неклассических задач (это — задачи, содержащие только ограничения типа $u \in U$ без условий на правом конце траектории; в [40] опубликовано решение задачи о максимальной дальности планирования) встретились определенные трудности, связанные с огра-

нениями фазовой траектории. Делались попытки преодолеть их методом штрафных функций, но затем основным вычислительным средством стали методы вариаций в фазовом пространстве. Разумеется, простота учета фазовых ограничений, являющихся наиболее сложными с теоретической точки зрения, не дается даром: реализация алгоритмов спуска в фазовом пространстве сталкивается с определенными трудностями в смысле объема вычислений и при учете ограничений типа $u \in U$. В своем месте это будет разъяснено.

Обоснование метода мы пачнем с обсуждения близкой, но все же существенно отличающейся от метода Н. Н. Моисеева, схемы приближенного решения задач оптимального управления. Имеется в виду популярное в теоретических исследованиях сведение к задаче *математического программирования*. Вводится сетка t_0, t_1, \dots, t_N ; уравнения, функционалы и ограничения заменяются соответствующими разностными аппроксимациями на сеточных функциях $\{x_i\}_{i=0}^N, \{u_{i+1/2}\}_{i=0}^{N-1}$. Так получаем задачу: найти сеточную траекторию из условий

$$\min \sum_{i=0}^{N-1} \tau f^0\left(\frac{x_i + x_{i+1}}{2}, u_{i+1/2}\right) \\ (\min F_0[x, u]), \quad (7)$$

$$\frac{x_{i+1} - x_i}{\tau} = f\left(\frac{x_i + x_{i+1}}{2}, u_{i+1/2}\right), \quad i = 0, 1, \dots, N-1,$$

$$x_0 = X_0, \quad x_N = X_1, \quad x_i \in G, \quad u_{i+1/2} \in U.$$

Сразу же возникает традиционный в математике вопрос о сходимости (при $N \rightarrow \infty, \tau = T/N \rightarrow 0$) сеточного решения задачи (7) к решению исходной задачи (1)–(5). Этот вопрос подробно исследован, например, в [14], однако ответ на него, в сущности, очевиден, да и само доказательство, если оставить в стороне стремление к чрезмерной общности и педантичное перечисление всех предположений, тоже не очень сложно. Конечно, эта простота связана в значительной мере с тем, что наиболее тонкие вопросы были решены до постановки задачи (7) в связи с другими проблемами. Мы приведем эскиз доказательства, постаравшись выделить наиболее важные содержательные моменты и предоставив читателю либо самому аккуратно оформить все «ε-δ»-формулировки, либо обратиться к [14]. Доказательству подлежат два факта.

1. Пусть $\{x^*(t), u^*(t)\}$ — оптимальная траектория дифференциальной задачи (1)–(5), а F_0^* — соответствующее значение функционала (1). Тогда при достаточно малых τ существуют и дискретные траектории $\{x_\tau^*, u_\tau^*\}$, для которых $F_0[x_\tau^*, u_\tau^*] < F_0^* + \eta(\tau)$, $\eta(\tau) \rightarrow 0$ при $\tau \rightarrow 0$. Заметим, что $\{x_\tau^*, u_\tau^*\}$ не является, вообще говоря, решением задачи (7), но $F_0[x_\tau^*, u_\tau^*]$ мажорирует значение функционала

в задаче (7). Такую сеточную траекторию легко построить для системы с выпуклой вектограммой расширенной системы, т. е. предположив выпуклость множества $\{f^0(x, u), f(x, u)\}$, $u \in U$ при всех x . Тогда в качестве сеточной управляющей функции можно взять точки $u_{i+1/2}$, вычисленные следующим образом: находится точка

$$f_{i+1/2} = \frac{1}{\tau} \int_{t_i}^{t_{i+1}} f[x(t_i), u(t)] dt \in \text{conv } f(x, U).$$

В силу выпуклости существует точка $u_{i+1/2} \in U$ такая, что $f_{i+1/2} = f[x(t_i), u_{i+1/2}]$. Обычные оценки, используемые при обосновании методов численного интегрирования обыкновенных дифференциальных уравнений, позволяют утверждать, что решение разностной системы $x_{i+1} = x_i + \tau f(x_i, u_{i+1/2})$ аппроксимирует траекторию $x^*(t)$, $\|x_i - x^*(t_i)\| \leq C\tau$, причем постоянная C зависит только от длины интервала T и константы условия Липшица для функции $f(x, u)$: $\|f(x, u) - f(x', u)\| \leq C_1 \|x - x'\|$ (это условие, разумеется, нужно оговорить). Теперь следует ослабить формулировку разностной задачи (7), потребовав выполнения условий $x_i \in G$, $x_N = X_1$ лишь с точностью до $C_2\tau$ (или с точностью до $\sqrt{\tau}$), с тем, чтобы построенная выше разностная траектория могла считаться допустимым решением разностной задачи (7), а для решения этой задачи, существование которого следует из элементарных теорем о достижении минимума в конечномерных пространствах, получаем оценку минимизируемого функционала сверху:

$$\min_{u_\tau} F_0^{(\tau)} \leq F_0^* + O(\tau).$$

Таким образом, рассматривая разностные задачи для последовательности $N \rightarrow \infty$ ($\tau \rightarrow 0$), получим

$$\lim_{\tau \rightarrow 0} \min_{u_\tau} F_0[u_\tau] \leq F_0^*.$$

2. Теперь следует доказать обратное неравенство: $\lim_{\tau \rightarrow 0} \min_{u_\tau} F_0[u_\tau] \geq F_0^*$. Пусть, напротив, существует последовательность $\tau_1 > \tau_2 > \dots \rightarrow 0$, для которой решения разностных задач с шагом τ_k дают значения функционалов $F_0\{\tau_k\} < F_0^* - \alpha$, $\alpha > 0$. Каждую сеточную траекторию дополним до непрерывной функции $x^{(k)}(t)$ линейной интерполяцией. Тогда функции $x^{(k)}(t)$ удовлетворяют условиям $x^{(k)}(t) \in G$, $x^{(k)}(T) = X_1$ с точностью до $O(\tau_k)$ и образуют компактное в C семейство. В этом случае существует предельная функция $x(t)$, почти всюду удовлетворяющая дифференциальному уравнению (2), удовлетворяющая условиям (3)–(5) и доставляющая функционалу (1) значение, не большее $F_0^* - \alpha$, что противоречит предположению минимальности F_0^* . Таким об-

разом доказано, что решение разностной вариационной задачи по функционалу не хуже (с точностью до $O(\tau)$) решения дифференциальной и не может быть существенно лучше. Это и означает, что разностная задача аппроксимирует дифференциальную. Этим мы и ограничимся, заметив, что строгое доказательство в основном следует этой схеме рассуждений, но в достаточно общем случае (невыхуклая вектограмма, зависимость U и G от t , и т. п.) требует довольно громоздкого, хотя в основном и чисто технического, оформления.

Перейдем к вопросам сходимости в вычислительной схеме Н. Н. Моисеева. Основное осложнение связано с тем, что теперь в разностной задаче (7) точки x_i могут принимать лишь дискретные значения x_j^i , принадлежащие сетке S^i . Поэтому в принципе может оказаться, что ни для какой пары точек из соседних сеток $\{x_j^i, x_k^{i+1}\}$ не удастся построить соединяющей их траектории (1) на малом интервале $[t_i, t_{i+1}]$. В этом случае разностная задача просто не имеет решения. Чтобы избежать этой опасности, следует наложить определенные ограничения на h -шаг сетки по фазовым координатам. Кроме того, нужно гарантировать разрешимость элементарной операции. Эти вопросы исследовались в работах [56], [37]. Разрешимость разностной задачи и сходимости численного решения к решению задачи (1)–(5) была доказана в предположении некоторых свойств непрерывности функции Беллмана решаемой задачи. Однако для практики вычислений более существенным является другое условие: шаги сетки h_r по r -й компоненте фазового пространства должны быть связаны с шагом сетки по времени τ соотношением $h_r = \tau^{1+p_r}$, где $p_r \geq 1$ — некоторые числа, зависящие от строения области достижимости за малое время τ для системы (1). Напомним, что *областью достижимости* $D(Z, \tau)$ называется совокупность правых концов траекторий системы $\dot{x} = f(x, u)$, $x(0) = z$ при произвольных измеримых $u(t)$, $u(t) \in U$, $0 \leq t \leq \tau$. В работе автора [93] те же вопросы были решены только с одним предположением $h = O(\tau^2)$. При этом под элементарной операцией следует понимать решение следующей простой геометрической задачи, являющейся аппроксимацией дифференциальной на малом интервале времени. Для расширенной системы (1) (пополненной уравнением $\dot{x}^0 = f^0(x, u)$, $x^0(0) = 0$) строится в каждой точке x область $x + \tau f(x, U)$ (если $f(x, U)$ не выпукла, следует заменить ее выпуклой оболочкой). Далее эта область расширяется присоединением всех сфер радиуса $C\tau^2$ с центрами в $x + \tau f(x_1, U)$. Полученную область в пространстве $\{x^0, x^1, \dots, x^n\}$ обозначим $D_\tau(x)$, а ее проекцию на гиперплоскость $\{x^1, x^2, \dots, x^n\}$ — $D_\tau^*(x)$. Если шаги сеток $h = c\tau^2$, то при определенном соотношении между c и C можно утверждать, что для любой точки $x_j^i \in S^i$ найдется хотя бы одна точка $x_k^{i+1} \in S^{i+1}$ такая, что

$x_k^{i+1} \in D_\tau^*(x_j^i)$, и если существует траектория задачи (1)—(5), соединяющая X_0 с X_1 , то существует и траектория разностной задачи, проходящая через узлы сеток S^i . Далее, в качестве оптимальной стоимости перехода из x_j^i в x_k^{i+1} можно взять решение задачи

$$\delta F_{j,k}^{i+1/2} = \min x^0 \text{ при условии } \{x^0, x_k^{i+1}\} \in D_\tau(x_j^i).$$

Заметим, что при таком определении элементарной операции сеточная траектория является аппроксимацией с точностью до $O(\tau)$ какой-то траектории системы (2). Функционал (7) также аппроксимирует (4) с точностью $O(\tau)$. Кстати, можно брать и $h = c\tau^{1+\varepsilon}$, $1 > \varepsilon > 0$, но в этом случае аппроксимация имеет порядок $O(\tau^\varepsilon)$.

Стоит разъяснить возможное в связи с этим недоразумение. Хорошо известно, что область достижимости за время τ для управляемой системы обычно имеет существенно разные размеры по разным координатам. Так, для системы

$$\dot{x}^1 = x^2; \quad \dot{x}^2 = x^3; \quad \dot{x}^3 = u, \quad |u| \leq 1$$

область достижимости за время τ имеет размер $O(\tau)$ по x^3 , $O(\tau^2)$ по x^2 , $O(\tau^3)$ по x^1 . В предложенной же выше реализации элементарной операции область достижимости по всем координатам имеет размер не менее $O(\tau^2)$. Не является ли это недопустимым расширением технических возможностей системы? Другими словами, можно ли гарантировать, что любая ломаная Эйлера, $x_{i+1} \in D_\tau(x_i)$, $i = 0, 1, \dots, N-1$, аппроксимирует (с точностью до $O(\tau)$) какую-то траекторию системы (2). Это действительно можно доказать, предположив, разумеется, выполнение условия Липшица по x для функции $f(x, u)$. Мы не будем приводить этого очень простого, в сущности, доказательства, но сошлемся для разъяснения на хорошо известный факт: для «неуправляемой» системы $\dot{x} = f(x)$ область достижимости за время τ есть точка. Однако ломаная Эйлера $x_{i+1} = x_i + \tau f(x_i) + O(\tau^2)$ сходится со скоростью $O(\tau)$ к траектории системы $\dot{x} = f(x)$ при любой величине $O(\tau^2)$, лишь бы она равномерно оценивалась: $\|O(\tau^2)\| \leq C\tau^2$.

Условие на шаг $h = O(\tau^2)$ неприятно, так как с ним связан большой объем вычислений. Ослабить его и заменить соотношением $h = O(\tau)$ в принципе нельзя. Это может привести (и в простых примерах действительно приводит) к тому, что сеточные оптимальные траектории не сходятся (при $h, \tau \rightarrow 0$, $h/\tau = \text{const}$) к решению исходной задачи (1)—(5). Этот факт нетрудно понять, пользуясь простыми качественными соображениями. В самом деле, при $h = \tau$ множество сеточных траекторий (т. е., например, кусочно линейных функций, проходящих через узлы сеток S^i) образует в пространстве непрерывных функций h -сеть, если в качестве нормы рассматривать величину $\|x(\cdot)\| = \max_t \|x(t)\|$. Однако в простран-

стве пар $\{x(t), \dot{x}(t)\}$ это множество уже при $h \rightarrow 0$ плотной сети не образует, так как \dot{x} принимает только целые значения. Поскольку управление u более или менее соответствует производной

\dot{x} , то и в пространстве $\{x(t), u(t)\}$ множество сеточных траекторий не плотно при любом τ . Этим мы закончим обсуждение теоретических вопросов, возникающих в связи со схемой Н. Н. Моисеева.

§ 16. Метод вариаций в фазовом пространстве. Вычислительные схемы

Метод вариаций в фазовом пространстве в той форме, которая была подробно описана и исследована в предыдущем параграфе, не реализуем на современных ЭВМ даже для самых простых задач с размерностью фазового пространства $n > 2$. Однако на его основе было разработано несколько практических алгоритмов итерационного типа, которые использовались для фактического решения реальных прикладных задач. Ниже эти алгоритмы будут описаны в общих чертах и обсуждены с точки зрения сходимости полученных с их помощью численных решений к решению исходной задачи.

Метод локальных вариаций. Метод, разработанный Ф. Л. Черноусько, представляет собой, видимо, наиболее широко используемую форму метода вариаций в фазовом пространстве. Метод носит итерационный характер, каждая итерация является переходом от некоторой траектории к близкой к ней, лучшей по величине минимизируемого функционала. Пусть $x(t)$ — некоторая траектория системы $\dot{x}=f$, удовлетворяющая краевым условиям $x(0)=X_0$, $x(T)=X_1$ и фазовым ограничениям. Эту траекторию можно представить последовательностью точек на временной сетке

$$X_0 = x(t_0), x(t_1), \dots, x(t_i), \dots, x(t_N) = X_1, \quad (1)$$

причем переход из точки $x(t_i)$ в $x(t_{i+1})$ осуществляется «элементарной операцией» и стоимость его — число $\delta F^{i+1/2}[x(t_i), x(t_{i+1})]$. Таким образом, на траектории (1) функционал имеет значение

$$F = \sum_{i=0}^{N-1} \delta F^{i+1/2}[x(t_i), x(t_{i+1})]. \quad (2)$$

Пусть i — некоторое целое число, $0 < i < N$; рассмотрим траектории типа (1), отличающиеся от «опорной» траектории (1) только значением x в точке t_i , а именно: рассматриваются смещенные положения точки $x(t_i)$: $x(t_i) \pm h_k e_k$, $k=1, 2, \dots, n$, где e_k — k -й орт в n -мерном пространстве, h_k — шаг по k -й компоненте фазового вектора. В сумме (2) при этом меняются только два слагаемых — $\delta F^{i-1/2}$ и $\delta F^{i+1/2}$. Перебрав таким образом $2n$ вариантов*), находим лучшую

*) Иногда перебираются не все $2n$ вариантов, а переход к новой траектории осуществляется, как только в процессе перебора встретится лучшая траектория.

траекторию со смещенным положением $x(t_i)$ (рис. 12) (в принципе положение $x(t_i)$ может и не измениться). Изменив $x(t_i)$ и сумму (2), проделываем ту же самую операцию над точкой $x(t_{i+1})$ и т. д. Улучшив исходную траекторию подобными вариациями последовательно точек $x(t_1)$, $x(t_2)$, ..., $x(t_{N-1})$, получаем новую траекторию.

Это и есть основной цикл метода локальных вариаций; его временная стоимость $\sim 2nN$ элементарных операций. Рассмотрим связанные с этим методом вычислительные трудности и некоторые способы их преодоления.

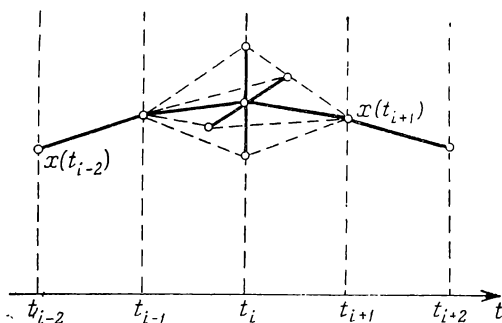


Рис. 12.

Построение элементарной операции. В принципе, можно использовать конструкцию конечно-разностного типа

$$x_{i+1} = x_i + \tau f\left(\frac{x_i + x_{i+1}}{2}, u\right), \quad x_i \equiv x(t_i), \quad (3)$$

и, задав x_i , x_{i+1} , искать $u_{i+1/2}$ решением задачи

$$\min_{u \in U} f^0\left(\frac{x_i + x_{i+1}}{2}, u\right) \quad (4)$$

при условиях (3). Однако в (3) мы имеем n условий, в то время как размерность управления u обычно меньше n , и в общем случае условия (3) просто несовместны: если размерность u равна n , то условия (3) определяют $u_{i+1/2}$ однозначно (с формальной точки зрения, опирающейся лишь на подсчет степеней свободы u и уравнений (3)). Если при этом окажется $u_{i+1/2} \notin U$, элементарная операция реализована, однако нет никаких оснований утверждать, что обязательно $u \in U$. Для преодоления затруднений, связанных с недостатком степеней свободы в u , были предложены некоторые приемы. Рассмотрим и обсудим их.

Метод дробных шагов. Интервал (t_i, t_{i+1}) разбивается на k равных частей, (3) и (4) заменяются на

$$x_{i+1} = x_i + \sum_{j=1}^k \frac{\tau}{k} f\left(\frac{x_i + x_{i+1}}{2}, u_j\right), \quad u_j \in U, \quad (3^*)$$

$$\min_{\{u_j\}} \sum_{j=1}^k \frac{\tau}{k} f^0\left(\frac{x_i + x_{i+1}}{2}, u_j\right), \quad (4^*)$$

За счет k можно обеспечить в (3^*) избыток степеней свободы по сравнению с числом условий n , и формально задача (3^*) — (4^*) становится обычной задачей на условный минимум. Однако наличие в (3^*) малого параметра τ и ограничения $u_j \in U$ делают такой формальный подход недостаточным. В самом деле, даже используя на (t_i, t_{i+1}) произвольную функцию $u(t) \in U$, мы получим вместо (3^*) условие

$$x_{i+1} = x_i + \int_{t_i}^{t_{i+1}} f\left(\frac{x_i + x_{i+1}}{2}, u(t)\right) dt, \quad (3^{**})$$

которое может оказаться (и, как правило, оказывается) неразрешимым. Дело в том, что область *)

$$x_i + \tau f\left(\frac{x_i + x_{i+1}}{2}, U\right)$$

часто имеет размерность ниже размерности фазового пространства, и поскольку в качестве возможных новых положений точки x_{i+1} берется крайне редкое множество $x_{i+1} + h e_k$, $k=1, 2, \dots, n$, построенное к тому же без учета структуры и расположения в пространстве области $x + \tau f(x, U)$, разрешимость (3^*) или (3^{**}) далеко не очевидна и в общем случае доказана быть не может.

Метод бегущей волны. В работе [17] предложен прием, позволяющий, видимо, преодолеть трудности, связанные с малой размерностью u . Именно, предлагается решать задачу типа (3)—(4) не на одном интервале (t_i, t_{i+1}) , а на k интервалах (t_i, t_{i+1}) , (t_{i+1}, t_{i+2}) , \dots , (t_{i+k-1}, t_{i+k}) одновременно:

$$\min \tau \sum_{j=i}^{i+k-1} f^0\left(\frac{x_j + x_{j+1}}{2}, u_{j+1/2}\right) \quad (5)$$

при условиях

$$x_{j+1} = x_j + \tau f\left(\frac{x_j + x_{j+1}}{2}, u_{j+1/2}\right), \quad j = i, i+1, \dots, i+k-1, \quad (6)$$

*) Напомним, что $f(x, U)$ есть выпуклая оболочка совокупности точек $\{f(x, u)\}$ при всех $u \in U$.

причем свободными параметрами, за счет которых удовлетворяются kn условий (6) и минимизируется форма (5), являются $x_{i+1}, x_{i+2}, \dots, x_{i+k-1}, u_{i+1/2}, u_{i+1 1/2}, u_{i+k-1/2}$ (т. е. $(k-1)n + kr$ степеней свободы); x_i, x_{i+k} — фиксированы, а для $x_{i+1}, \dots, x_{i+k-1}$ уже нет характерного условия — принадлежности к некоторой редкой сетке; они, в принципе, могут быть какими угодно. Разумеется, это существенно осложняет задачу (5) — (6) по сравнению с задачей типа (3) — (4). В методе бегущей волны последовательно изменяются положения группы точек $\{x_1, x_2, \dots, x_{k-1}\}$, затем группы $\{x_2, x_3, \dots, x_k\}$ и т. д. до группы $\{x_{N-k+1}, \dots, x_{N-1}\}$. Это и есть основной итерационный цикл, который затем снова повторяется до стабилизации численного решения. Заметим, что в этом методе в значительной мере утрачена особая роль сеток в фазовом пространстве; их, в сущности, уже и не осталось, в то время как в методе локальных вариаций сетки в фазовом пространстве, пусть очень редкие (состоящие в каждой элементарной операции из одной-единственной точки $x(t_j)$ при $j \neq i$ и из $2n+1$ точек при $j = i$), все же сохраняются. Подобная эволюция метода вариаций в фазовом пространстве представляется нам весьма знаменательной; она связана с тем, что для задач оптимального управления выбор в качестве независимого функционального аргумента именно управления $u(t)$ является гораздо более естественным, чем выбор в качестве такового фазовой траектории $x(t)$.

Медленная сходимость. В § 15 было выяснено, что шаг h сетки в фазовом пространстве должен быть существенно меньше шага по времени τ , например, $h = O(\tau^2)$. Одна итерация метода локальных вариаций смещает исходную траекторию на расстояние h и для того, чтобы «добраться» до оптимальной траектории, следует совершить не менее $O\left(\frac{1}{h}\right) = O\left(\frac{1}{\tau^2}\right) \approx O(N^2)$ таких итераций. Таким образом, стоимость решения задачи $\sim O(nN^3)$ элементарных операций. В реальных задачах это довольно большое число, и следует заботиться о его сокращении.

В процессе эксплуатации метода соответствующие приемы были разработаны, и практическая технология выглядит следующим образом: сначала интервал $[0, T]$ разбивается на небольшое число (скажем $N=10$) частей, и шаг h достаточно велик. Итерации метода локальных вариаций повторяются до тех пор, пока они сопровождаются падением суммы (2). Как только встретились ситуация, в которой ни одно из $x(t_i)$ не было смещено с уменьшением (2), шаг h делится пополам, и с новым шагом h на той же временной сетке снова начинаются итерации. Если после уменьшения h первая же итерация не привела к вариации траектории, сетка по времени дробится, например, число N увеличивается вдвое. После этого шаг h увеличивается, и начинается очередной

цикл минимизации. Такая технология позволяет существенно сократить объем вычислений, если с большими τ и h получается достаточно хорошее начальное приближение.

Существенная неполнота локальных вариаций и тупиковые ситуации. Наиболее серьезным дефектом метода локальных вариаций является то, что он использует чрезвычайно узкое множество соседних с данной траекторий. В этом множестве может не оказаться лучшей, однако это не обязательно свидетельствует об оптимальности данной траектории и может быть следствием того, что алгоритм исследует не все возможные вариации траектории. Пример подобного рода строится очень просто; в задаче о вертикальном подъеме ракеты, подробно описанной в §§ 28, 29, ищется скалярная управляющая

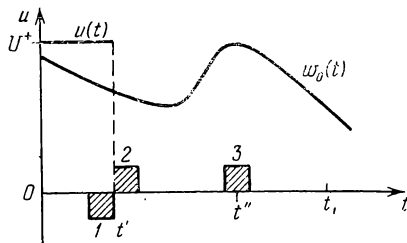


Рис. 13.

функция $u(t)$, $0 \leq u \leq U^+$, с целью максимизировать $F_0[u(\cdot)]$ (высоту в заданный момент времени $t=T$) при заданном расходе топлива: $\int_0^T u(t) dt = 0,8$. Пусть в качестве исходной траектории берется траектория, порождаемая управлением

$$u(t) = \begin{cases} U^+ & \text{при } 0 \leq t \leq 0,8/U^+, \\ 0 & \text{при } t > 0,8/U^+. \end{cases}$$

Тогда функциональная производная $\frac{\partial F_0[u(\cdot)]}{\partial u(\cdot)} = w_0(t)$ имеет вид, качественно изображенный на рис. 13. (Этот график, по существу, взят из расчетов, однако, для большей наглядности утрирован: в действительности колебания $w_0(t)$ на интервале $0 \leq t \leq t_1$ значительно меньше, но соотношения типа $\frac{d}{dt} w_0(t') < 0$, $w_0(t'') > w_0(t')$ на графике соответствуют действительной функции $w_0(t)$).

Рассмотрим малые вариации управления, составленные из трех изображенных на рисунке элементов.

Условия $0 \leq u + \delta u \leq U^+$ и $\int_0^T \delta u(t) dt = 0$ допускают только две комбинации:

1. Локальная вариация (1) + (2). Для нее

$$\delta F_0 = \int_0^T w_0(t) \delta u(t) dt < 0,$$

и она не улучшает данную траекторию.

2. Нелокальная вариация (1) + (3). Теперь

$$\delta F_0 = \int_0^T w_0(t) \delta u(t) dt > 0,$$

т. е. траектория очевидным образом неоптимальна, однако это можно обнаружить только в том случае, если рассматриваются нелокальные комбинации вариаций управления. Нетрудно показать, что в данном примере для любой вариации управления локального типа

$$\delta u(t) \begin{cases} \text{произвольна на интервале } (t^*, t^* + H), \\ = 0 \text{ вне } (t^*, t^* + H), \end{cases} \quad (7)$$

где t^* — произвольно, а $H > 0$ — некоторое не слишком большое число, одновременное выполнение условий

$$\int_0^T \delta u(t) dt = 0 \quad \text{и} \quad 0 \leq u(t) + \delta u(t) \leq U^+$$

неизбежно приводит к

$$\delta F_0[\delta u(\cdot)] = \int_0^T w_0(t) \delta u(t) dt \leq 0.$$

Другими словами, тривиально неоптимальная траектория оказывается оптимальной относительно неполного множества вариаций управления (7). Но методы локальных вариаций (включая сюда и метод бегущей волны) основаны на просмотре класса вариаций управления еще более узкого, чем класс (7), и подобные ситуации оказываются для них тупиковыми: траектория перестает варьироваться. Таким образом, сходимость этих методов доказана быть не может.

Сделанные выше утверждения нуждаются в пояснениях. Их не следует понимать в том смысле, что методом локальных вариаций нельзя получить решение задачи: ведь дефекты этого метода проявляются лишь в определенных ситуациях; в принципе, начав с некоторого достаточно разумного начального приближения, можно последовательными локальными вариациями получить траекторию, сколь угодно близкую к оптимальной, так и не столк-

нувшись с ситуацией, в которой метод не работает. Тем не менее метод локальных вариаций не является надежным, и, получив стабильную траекторию, вычислитель должен подвергнуть ее тщательному контролю, выяснив, в чем причина стабильности этой траектории: то ли дело в том, что она близка к оптимальной, то ли возникла тупиковая для метода локальных вариаций ситуация.

Метод трубки. Метод является упрощенным вариантом полного метода вариаций в фазовом пространстве. Это упрощение

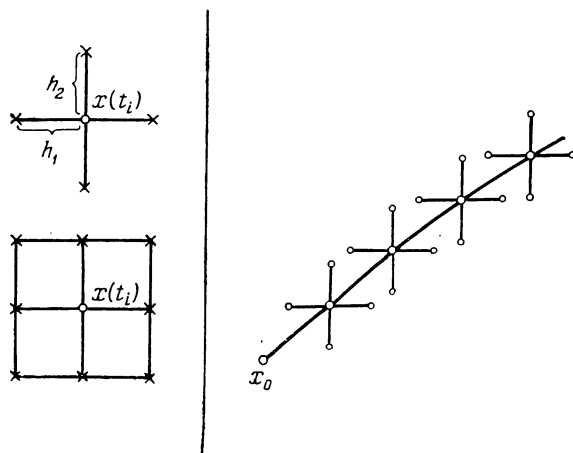


Рис. 14.

связано с тем, что в каждой точке t_i сетка S^i не покрывает допустимой области $G(t_i)$, а состоит из небольшого числа точек, соседних с точкой $x(t_i)$ данной опорной траектории. Например, сетка S^i может состоять из $2n+1$ точки (схема «крест»), или из 3^n точек (схема «решетка») (рис. 14). Если некоторые из точек такой решетки выходят за пределы $G(t_i)$, они, естественно, из S^i исключаются. В остальном вычислительная схема метода трубки совпадает с общей схемой. Вычисляются числа $\delta F^{i+1/2}$ для всевозможных пар точек из соседних сеток (ценой $N(2n+1)^2$ для «креста» или $N \cdot 3^{2n}$ для «решетки» элементарных операций), и методом динамического программирования находится наилучшая сеточная траектория; затем эта новая траектория становится опорной для построения нового набора сеток S^i и т. д. Тактика постепенного изменения шагов h и τ — та же, что и в методе локальных вариаций.

Вопросы обоснования этой вычислительной схемы в настоящее время остаются открытыми: в частности, не доказаны теоремы такого содержания:

I. Если некоторая траектория $x(t)$ методом трубки не варьируется (с учетом упомянутой выше тактики измельчения шагов h и τ), то она удовлетворяет необходимому условию оптимальности — принципу максимума.

II. Пусть $x(t)$ — предельная траектория для последовательности полученных методом трубки оптимальных траекторий $x(t, \tau_p)$, $\tau_1 > \tau_2 > \dots \rightarrow 0$, $x(t, \tau_p)$ — траектория, на которой остановился метод трубки при данном шаге временной сетки τ_p и при $h \rightarrow 0$ (эта траектория берется в качестве исходной для процесса с $\tau_{p+1} = \frac{1}{2} \tau_p$). Тогда $x(t)$ удовлетворяет принципу максимума, т. е. является в первом порядке стационарной.

Заметим, что выше сформулированы относительно слабые результаты: ясно, что от метода трубки, так же как и от всех практически реализуемых методов, нельзя требовать нахождения глобального минимума; метод заслуживает внимания и может быть использован, если для него стационарными траекториями (тупиковыми ситуациями) являются действительно стационарные, удовлетворяющие необходимому условию оптимальности — принципу максимума — траектории.

Можно ли доказать подобные теоремы для метода трубки при том бесхитроном способе построения сеток, который показан на рис. 14, или сетки следует строить с учетом структуры области достижимости для траектории $\dot{x}=f$ за малое время τ — неизвестно. Однако и контрпримера, аналогичного контрпримеру для метода локальных вариаций, насколько известно автору, не построено.

Сформулированные выше теоремы можно доказать, если сетки S' строить следующим образом: в окрестности $x(t_i)$ строится сфера радиуса $O(\tau)$ в пространстве $\{x\}$, и эта сфера покрывается сеткой с шагом $h=O(\tau^2)$, содержащей, таким образом $O(\tau^{-n})$ точек (вместо $O(\tau^{-2n})$ в полном методе). Однако такое упрощение, по существу, трудностей фактической реализации метода на современных ЭВМ не снимает.

Метод локальных вариаций и релаксационный метод. В [86] метод локальных вариаций был распространен на задачи минимизации функционалов от функций нескольких независимых переменных. Хорошо известно, что многие задачи математической физики (краевые задачи для уравнения Лапласа, для бигармонического уравнения и другие) могут быть сформулированы либо как задачи на минимум соответствующего функционала, либо как задачи с уравнениями в частных производных (эти уравнения — суть уравнения Эйлера для вариационной формулировки). Применительно к таким задачам метод локальных вариаций состоит из двух элементов.

1. Приближенное решение ищется в виде сеточной функции, а функционал заменяется соответствующей сеточной аппроксимацией. Например,

функционал

$$\int_0^1 \int_0^1 (u_x^2 + u_y^2) dx dy \quad (8)$$

после выбора числа N и введения в квадрате $0 \leq x, y \leq 1$ равномерной сетки с шагом $\Delta = 1/N$ и сеточной функции $u_{k,m} = u(k\Delta, m\Delta)$ аппроксимируется суммой

$$\sum_{k=0}^{N-1} \sum_{m=0}^{N-1} \Delta^2 \left\{ \left(\frac{u_{k+1,m} - u_{k,m}}{\Delta} \right)^2 + \left(\frac{u_{k,m+1} - u_{k,m}}{\Delta} \right)^2 \right\}. \quad (9)$$

2. Минимум сеточного функционала ищется процессом последовательного изменения значений сеточной функции в узлах, причем рекомендуется типичная для метода локальных вариаций технология: сначала делаются попытки менять каждую переменную на заданную величину h , они продолжаются (циклически) до тех пор, пока приводят к уменьшению функционала. Затем то же самое делается с шагом $h/2$, и т. д. Если отвлечься от этой технологии, то метод локальных вариаций, по существу, совпадает с хорошо известным *релаксационным методом*. Разница лишь в том, что в последнем смещение значения $u_{k,m}$ в узле сетки определяется решением задачи на минимум функционала, рассматриваемого в этот момент как функция только одного переменного $u_{k,m}$. Если функционал квадратичный, это смещение легко вычисляется явно. Для (9) это приводит к известной итерационной формуле (v — номер итерации):

$$u_{k,m}^{v+1} = \frac{1}{4} (u_{k-1,m}^{v+1} + u_{k+1,m}^{v+1} + u_{k,m-1}^v + u_{k,m+1}^v). \quad (10)$$

Таким образом, оба метода — суть некоторые варианты метода покоординатного спуска для минимизации (9). Следует иметь в виду, что на данном этапе основной проблемой в решении подобных задач является не столько построение аппроксимации типа (9), сколько разработка возможно более эффективных методов минимизации. Создание новой техники минимизации дает право говорить о новом методе решения задачи типа (8) — но лишь в том, разумеется, случае, если эта техника имеет какое-то преимущество по сравнению с уже известными. К сожалению, в публикациях по методу локальных вариаций (например, [41], [55], [56], [86]) нет данных, которые позволили бы оценить трудоемкость расчетов и сравнить с эффективностью стандартного релаксационного метода. К тому же сам по себе релаксационный метод в настоящее время относится к числу наиболее слабых, и при достаточно больших N (> 30) почти не употребляется. Вопросам ускорения процесса минимизации уделялось большое внимание; с некоторыми результатами по этому вопросу можно познакомиться по работам [16], [50], [24]. Здесь отметим лишь очень простое усовершенствование релаксационного метода — *метод последовательной сверхрелаксации*. После того как новое значение $u_{k,m}^{v+1}$ найдено из условия минимума функционала, оно еще раз пересчитывается по простой формуле

$$u_{k,m}^{v+1} := \omega u_{k,m}^{v+1} + (1 - \omega) u_{k,m}^v, \quad 1 < \omega < 2.$$

Для задачи (9) (и некоторых ее обобщений) теория выбора оптимального значения параметра релаксации ω построена, и известен эффект этого способа ускорения (см., например, [16], [50]). Так, для (9) разница между v -м приближением u^v и решением (точкой минимума (9)) стремится к нулю как $[1 - O(1/N^2)]^v$. Сверхрелаксация с оптимальным ω приводит к существенно более быстрой сходимости: $[1 - O(1/N)]^v$. Метод сверхрелаксации часто употребляют и в теоретически неисследованных ситуациях, назначая ω в диапа-

зоне, например, 1,5—1,7, и уточняя (в случае массовых расчетов) значение ω экспериментально. Неоднократно отмечалось заметное (в несколько раз по сравнению с обычной техникой релаксационного метода) сокращение числа итераций, обеспечивающих заданную точность.

§ 17. ϵ -метод Балакришнана

Метод, описание которого будет дано ниже, предложен пока для очень частного класса задач, и его фактическая реализация приводит к довольно громоздким и трудоемким вычислениям. Однако он заслуживает внимания, тем более, что его становление содержит поучительные моменты; ниже они будут особо подчеркнуты.

Итак, рассматривается задача классического типа:

$$\min_{u(\cdot)} F[u(\cdot)] \quad (1)$$

на траектории системы

$$\dot{x} = f(x, u), \quad x(0) = X_0, \quad 0 \leq t \leq T, \quad (2)$$

при заданных условиях на правом конце

$$x(T) = X_1. \quad (3)$$

Условие $u(t) \in U$ — отсутствует.

Основу метода составляет отказ от точного выполнения дифференциальных связей (2), и поиск минимума происходит в пространстве функций $\{x(\cdot), u(\cdot)\}$, рассматриваемых теперь как независимые аргументы задачи. Решение задачи осуществляется минимизацией функционала

$$F_\epsilon[u(\cdot), x(\cdot)] \equiv F_0[u(\cdot), x(\cdot)] + \frac{1}{\epsilon} \int_0^T \|\dot{x} - f(x, u)\|^2 dt, \quad (4)$$

причем на $x(t)$ накладываются условия: $x(0) = X_0$, $x(T) = X_1$, $\epsilon > 0$ — достаточно малое число. Поскольку в этом методе $x(\cdot)$ и $u(\cdot)$ — равноправны, естественно изменить стандартные обозначения для функционалов и включить $x(\cdot)$ в число функциональных аргументов. Мы не будем обсуждать вопросов обоснования метода, т. е. доказательства того интуитивно очевидного факта, что при достаточно малом ϵ решение задачи (4) сколь угодно близко аппроксимирует решение исходной задачи. Нас будет интересовать вычислительная сторона вопроса. Прежде всего заметим, что идея перехода к (4) не нова, и если бы все дело было в ней, не было бы никаких оснований считать Балакришнана автором метода. Дело в том, что сам по себе переход к (4) еще не образует метода, вычислительный метод появится лишь тогда, когда будет построен эффективный метод поиска минимума функционала F_ϵ . В работе

[78] предлагается искать решение $\{x(t), u(t)\}$ в форме конечных сумм по подходящей системе функций. Например,

$$u(t) = \sum_{k=0}^p \left(a_k \sin k\pi \frac{t}{T} + b_k \cos k\pi \frac{t}{T} \right),$$

$$x(t) = X_0 + \frac{t}{T} (X_1 - X_0) + \sum_{k=1}^q c_k \sin k\pi \frac{t}{T}.$$
(5)

После этого функционал (5) становится функцией конечного числа переменных $\{a_k, b_k, c_k\}_k$. Обозначим совокупность параметров $\{a_k, b_k, c_k\}$ через α ; размерность вектора α есть, очевидно, $2pr + +qn$, где r — размерность управления, n — размерность фазы x . Итак,

$$F_\epsilon[u(\cdot), x(\cdot)] = \Phi(\alpha). \quad (6)$$

Однако и сведение вариационной задачи (1)–(3) к конечномерной задаче минимизации $\Phi(\alpha)$ еще не дает метода, поскольку поиск минимума $\Phi(\alpha)$ оказывается чрезвычайно трудоемким и большие затраты машинного времени приводят к довольно ненадежным результатам. Причины этого подробно обсуждаются в § 25, здесь же заметим только, что при очень малом ϵ в функционале (4) основную роль играют невязки $\dot{x} - f(x, u)$, на фоне которых «теряется» исходный подлежащий минимизации функционал F_0 . Основной целью процесса поиска минимума $\Phi(\alpha)$ является минимизация $\|\dot{x} - f(x, u)\|$, и лишь после того как эта величина более или менее минимизирована, принимается во внимание значение F_0 . Другими словами, определяемая конструкцией (4) функция $\Phi(\alpha)$ оказывается очень негладкой, и для нее не удастся построить эффективный процесс минимизации. Именно с этим обстоятельством связана та довольно сложная и громоздкая конструкция поиска минимума $\Phi(\alpha)$, которая опирается на обширную информацию, включающую не только значения функции $\Phi(\alpha)$ и ее производных, но и значения производных отдельных составляющих $\Phi(\alpha)$ компонент.

Именно построение этой специализированной техники минимизации Φ и является основным оригинальным моментом, позволяющим связывать метод в целом с именем Балакришнана.

На отрезке $[0, T]$ вводится сетка t_1, t_2, \dots, t_N и определяется вектор невязок:

$$h_i(\alpha) \equiv \dot{x}(t_i) - f[x(t_i), u(t_i)], \quad i = 1, 2, \dots, N, \quad (7)$$

и функция $\Phi(\alpha)$ определяется аппроксимацией (4)

$$\Phi(\alpha) \equiv F_0[x(\cdot), u(\cdot)] + \frac{1}{\epsilon} \sum_{i=1}^N \|h_i(\alpha)\|^2. \quad (8)$$

Если конкретно F_0 определяется, например, формулой

$$F_0[u(\cdot), x(\cdot)] \equiv \int_0^T \Phi_0[u(t), x(t)] dt \quad (9)$$

с заданной функцией $\Phi_0(u, x)$, то интеграл заменяется той или иной аппроксимирующей суммой.

Основной цикл процесса минимизации $\Phi(\alpha)$ состоит из следующих операций:

1. Имея некоторое приближение α , вычисляем значения F_0 , $h_i(\alpha)$, $i=1, \dots, N$. Заметим, что трудоемкость этих вычислений примерно равноценна однократному численному интегрированию системы $\dot{x}=f$.

2. Обозначим $Z(\alpha) = \{F_0, h_1, h_2, \dots, h_N(\alpha)\}$, и вычислим матрицу

$$\frac{\partial Z}{\partial \alpha} = \left\{ \frac{\partial F_0}{\partial \alpha}, \frac{\partial h_1}{\partial \alpha}, \dots, \frac{\partial h_N}{\partial \alpha} \right\}. \quad (10)$$

Размерность вектора Z есть $Nn+1$, следовательно, объем памяти, необходимый для запоминания Z_α , исчисляется $(Nn+1) \times \times (2pr+qn)$ ячейками и достаточен даже в простых задачах. Вычисление матрицы Z_α осуществляется численным дифференцированием, что требует $(2pr+qn)$ -кратного вычисления вектора Z с варьируемыми последовательно компонентами α и «стоит», таким образом, $(2pr+qn)$ -кратного интегрирования системы $\dot{x}=f$.

3. Задача минимизации $\Phi(\alpha)$ линеаризуется в окрестности данной точки α , и следующее значение α ищется в форме $\alpha + \delta\alpha$, а для $\delta\alpha$ имеем задачу минимизации квадратичной формы

$$\min_{\delta\alpha} \left\{ F_0(\alpha) + \frac{\partial F_0}{\partial \alpha} \delta\alpha + \frac{1}{\epsilon} \sum_{i=1}^N \left\| h_i(\alpha) + \frac{\partial h_i}{\partial \alpha} \delta\alpha \right\|^2 \right\}. \quad (11)$$

Решение этой задачи может быть осуществлено методом сопряженных градиентов (см. § 51), или, если позволяют возможности машины, решением системы линейных уравнений

$$\frac{1}{2} \frac{\partial F_0}{\partial \alpha_j} + \frac{1}{\epsilon} \sum_{i=1}^N \left(h_i + \frac{\partial h_i}{\partial \alpha} \delta\alpha, \frac{\partial h_i}{\partial \alpha_j} \right) = 0, \quad j = 1, 2, \dots, (2pr + qn). \quad (12)$$

4. Вычисляется новое значение $\alpha := \alpha + \delta\alpha$, и далее процесс повторяется до получения необходимой точности.

З а м е ч а н и е 1. Включить в алгоритм учет условий на управление типа, например, $\varphi(u) \leq 0$, фазовых ограничений $G(x) \leq 0$, и ограничений общего вида $R(x, u) \leq 0$, в принципе,

можно двумя способами: либо методом штрафных функций, искажив функционал F_0 штрафной добавкой:

$$\bar{F}_0 \equiv F_0 + \frac{1}{\epsilon_1} \int_0^x [R^+(x, u)]^2 dt, \quad (13)$$

где $R^+ = \frac{1}{2}(R + |R|)$, а ϵ_1 — достаточно малое число, либо введя аналогичный вектору невязок $h_i(\alpha)$ вектор условий $r_i(\alpha) \equiv R[x(t_i), u(t_i)] \leq 0$, после чего задача (11) осложняется условиями:

$$r_i(\alpha) + \frac{\partial r_i}{\partial \alpha} \delta \alpha \leq 0, \quad i = 1, 2, \dots, N. \quad (14)$$

Первый способ не требует никаких изменений в алгоритмической схеме, однако он противоречит основной идее метода: формальное введение конструкций типа (13) портит дифференциальные свойства минимизируемой функции $\Phi(\alpha)$, и, если не принять специальных мер, делает метод неэффективным. Второй способ вполне укладывается в общую идеологию метода, но приводит к увеличению объема матрицы влияния (трудоемкость ее вычисления, однако, по существу, не меняется) и осложняет процесс определения $\delta \alpha$ решением задачи на условный минимум (11)–(14); сведение к системе линейных уравнений (12) уже, в частности, не проходит.

З а м е ч а н и е 2. Выше для простоты мы использовали в конечных рядах (5) базисные функции $\sin x$, $\cos x$; разумеется, это не связано с существом дела, и в конкретных задачах, характер решений которых качественно известен, могут быть выбраны наиболее подходящие базисные функции, что позволит решать задачу с небольшим числом их. Возникает и вопрос о разумном соотношении размерности аппроксимирующего пространства (числа членов в суммах (5)) с числом точек сетки N .

В работе [78] приведен пример решения описанным выше методом следующей задачи: найти $\min_{T, u(\cdot)} T$ на траектории системы

$$\begin{aligned} \dot{x}^1 &= T[f(x^1, x^2) - g \sin u], & \dot{x}^2 &= T x^1 \sin u, & 0 \leq t \leq 1; \\ x(0) &= X_0, & x(1) &= X_1. \end{aligned}$$

За 8 итераций в 11 секунд на CDC-6600 было получено стабильное решение. Точность выполнения уравнений $\dot{x}=f$ была проконтролирована интегрированием задачи Коши с найденным $u(t)$ и оказалась хорошей. Если бы такой же контроль был проведен и в расчетах [77], их ошибочность немедленно обнаружилась бы. В [78] не приведены значения p, q, N . Вид f также не сообщается, и расчет не удастся повторить другим методом.

§ 18. Метод проекции градиента

Задача оптимального управления, сколь бы ни была она сложна, часто допускает очень простую и изящную формулировку: найти точку функционального пространства u из условия

$$\min_{u \in \Omega} F_0(u), \quad (1)$$

где Ω — некоторая область функционального пространства, а F_0 — функционал, дифференцируемый по Фреше. Производную F'_0 обозначим, как обычно, через w_0 . Тогда:

$$F_0(u + \delta u) = F_0(u) + (w_0, \delta u) + O(\|\delta u\|^2). \quad (2)$$

Для решения задачи (1) давно предложен и обоснован (при определенных предположениях, обсуждать которые мы здесь не станем) метод проекции градиента. Он представляет собой алгоритм построения минимизирующей последовательности точек u . Пусть имеется некоторое u^k . Тогда в качестве следующего приближения берется точка

$$u^{k+1} = P_\Omega(u^k - Sw_0^k). \quad (3)$$

Здесь S — скалярный параметр, шаг процесса, w_0^k — градиент $F_0(u)$, вычисленный в точке u^k , и, наконец, P_Ω — оператор проектирования на множество Ω . Сама операция P_Ω определяется очень просто: для вычисления $P_\Omega(z)$ требуется решить задачу нахождения

$$\min_{y \in \Omega} \|y - z\| = \|z^* - z\|, \quad (4)$$

и тогда, по определению, $P_\Omega z \equiv z^*$. Для того чтобы алгоритм был полностью детерминирован, осталось указать способ определения S . Но это уже совсем несложно: определим однопараметрическое семейство точек $u(s) = P_\Omega(u^k - sw_0^k)$ и решим задачу одномерной минимизации

$$\min_s F_0[u(s)] = F_0[u(S)]. \quad (5)$$

Попробуем использовать эту общую схему в задаче оптимального управления. При этом встретятся определенные трудности, связанные с фактической реализацией таких принципиально несложных операций, как, например, проектирование.

Итак, рассмотрим задачу: найти

$$\min F_0[u(\cdot)] \quad (6)$$

при условиях

$$\left. \begin{array}{l} 1) \quad \dot{x} = f(x, u), \quad \Gamma(x) = 0, \quad 0 \leq t \leq T. \\ 2) \quad u(t) \in U, \quad t \in [0, T]. \\ 3) \quad F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m. \end{array} \right\} \quad (7)$$

Все функционалы F_i будем полагать дифференцируемыми в смысле Фреше. Прежде всего заметим, что есть по крайней мере два способа формулировать задачу (6)—(7) в виде (1).

1. В качестве элемента u абстрактной постановки задачи (1) можно взять измеримую вектор-функцию $u(\cdot)$, а множество Ω выделяется условиями: $u(t) \in U$ при всех t и $F_i[u(\cdot)] = 0$, $i = 1, 2, \dots, m$. При этом конкретные формулы, определяющие F_i через $u(\cdot)$, содержат еще и фазовые координаты $x(t)$, однозначно определяемые краевой задачей $\{\dot{x} = f; \Gamma(x) = 0\}$.

Вычисление производной $\partial F_0[u(\cdot)]/\partial u(\cdot) = w_0(\cdot)$ осуществляется известным образом и особых трудностей не содержит. Сложнее с проектированием. Оно сводится при фиксированном значении параметра s к следующей задаче: найти

$$\min_{v(\cdot)} \int_0^T \|v(t) - u^k(t) + sw_0^k(t)\|^2 dt \quad (8)$$

при условиях

$$v(t) \in U \quad \text{при всех } t \in [0, T], \quad (9)$$

$$F_i[v(\cdot)] = 0, \quad i = 1, 2, \dots, m, \quad (10)$$

причем формулы для F_i содержат фазовые координаты x , связанные с v краевой задачей $\{\dot{x} = f(x, v), \Gamma(x) = 0\}$. Таким образом, проектирование оказалось вариационной задачей, отличающейся от исходной только выражением для минимизируемого функционала. Эта задача в общем случае ничуть не легче исходной, а если еще учесть необходимость неоднократного решения ее с разными s для определения шага процесса S , то едва ли здесь можно говорить о каком-то продвижении в приближенном решении задачи. Скорее наоборот, приближенное решение задачи оптимального управления сведено к последовательности задач такой же, примерно, трудности.

2. Второй способ интерпретировать задачу (6)—(7) как абстрактную задачу (1) состоит в том, что элементом u формулировки (1) считается пара функций $\{u(\cdot), x(\cdot)\}$. В этом случае градиент F_0 вычисляется элементарным варьированием определяющей его значения формулы. Так, если

$$F_0[u(\cdot), x(\cdot)] \equiv \int_0^T \Phi[x(t), u(t)] dt,$$

то градиент есть

$$w(t) = \{\Phi_x[x(t), u(t)], \Phi_u[x(t), u(t)]\}.$$

Множество Ω в пространстве пар функций $\{u(\cdot), x(\cdot)\}$ выделяется условиями:

$$\dot{x} - f(x, u) = 0; \quad \Gamma(x) = 0; \quad u(t) \in U, \quad F_i[u(\cdot), x(\cdot)] = 0.$$

В основу дальнейшего будет положен первый способ, хотя некоторые алгоритмы (они будут описаны) основаны на втором. Выше мы убедились, что проектирование на Ω практически осуществить не удастся. Однако можно построить алгоритм, основанный на проектировании не на Ω , а на некоторое многообразие, которое условно можно считать касательным к Ω в точке u . Речь идет о следующем: линеаризуем задачу в окрестности данной точки u и построим метод проекции градиента для линеаризованной задачи. Пусть $w_i(t)$ — функциональные производные входящих в задачу функционалов $F_i[u(\cdot)]$, $i=0, 1, \dots, m$. Будем искать вариацию управления $\delta u(\cdot)$, решая вариационную задачу: найти

$$\min_{\delta u(\cdot)} \int_0^T \|Sw_0(t) + \delta u(t)\|^2 dt \quad (11)$$

при условиях

$$\delta u(t) \in \delta U(t), \quad (12)$$

$$F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt = 0, \quad i = 1, 2, \dots, m. \quad (13)$$

Здесь, как обычно, $\delta U(t)$ — некоторая окрестность точки $u(t)$, причем из $\delta u(t) \in \delta U(t)$ следует $u(t) + \delta u(t) \in U$. Кроме того, $\delta U(t)$ должна обеспечивать нужную точность линейного приближения и обладать свойством полноты (подробнее см. § 20). Заметим, что вместо условия (12) можно было бы использовать и более простое, не требующее дополнительных построений, условие

$$u(t) + \delta u(t) \in U, \quad (12^*)$$

поскольку необходимую малость вариации $\delta u(t)$ можно обеспечить соответствующим выбором параметра S . Мы предпочтем явное введение ограничивающих $\delta u(t)$ условий типа (12). Это связано с используемым в дальнейшем алгоритмом решения задачи (11)—(13); не исключено, что, используя другие алгоритмы, можно заменить (12) на (12*). Кроме того, нам будет удобно заменить минимизируемую форму (11) на эквивалентную

$$\min_{\delta u(\cdot)} \int_0^T \left\{ w_0(t) \delta u(t) + \frac{1}{2S} \|\delta u(t)\|^2 \right\} dt. \quad (11^*)$$

Задача (11)–(13) в принципе решается, и возможный алгоритм ее решения не так уж сложен, однако сам метод в целом уже не имеет той степени обоснованности, которой обладает метод проекции градиента в точной постановке: ведь мы не учитываем связанных с нелинейностью задачи величин, имеющих формально порядок $O(\|\delta u\|^2)$. Следующие простые теоремы содержат «грубое обоснование» метода, основанного на линеаризации.

Т е о р е м а 1. Пусть управление $u(\cdot)$ допустимо, т. е. $F_i[u(\cdot)] = 0$, $i = 1, 2, \dots, m$, $u(t) \in U$ при всех t , и пусть решение задачи (11)–(13) $\delta u^*(t) \neq 0$. Тогда $\delta u^*(t)$ есть улучшающая вариация управления в том смысле, что

$$1) \quad \delta F_0[\delta u^*(\cdot)] = \int_0^T w_0(t) \delta u^*(t) dt < 0.$$

$$2) \quad u(t) + \delta u^*(t) \in U.$$

$$3) \quad \delta F_i[\delta u^*(\cdot)] = \int_0^T w_i(t) \delta u^*(t) dt = 0, \quad i = 1, \dots, m.$$

Доказательство. Так как функция $\delta \tilde{a}(t) \equiv 0$ удовлетворяет условиям (12), (13) (ведь, по предположению, $F_i = 0$), то

$$\int_0^T \left\{ w_0 \delta u^* + \frac{1}{2S} \|\delta u^*\|^2 \right\} dt \leq \int_0^T \left\{ w_0 \delta \tilde{a} + \frac{1}{2S} \|\delta \tilde{a}\|^2 \right\} dt = 0.$$

Следовательно,

$$\int_0^T w_0(t) \delta u^*(t) dt \leq -\frac{1}{2S} \int_0^T \|\delta u^*(t)\|^2 dt < 0.$$

Следующая теорема утверждает, что если при решении задачи (11)–(13) будет получена вариация $\delta u^*(t) \equiv 0$, то данная траектория $\{u(\cdot), x(\cdot)\}$ удовлетворяет принципу максимума.

Т е о р е м а 2. Пусть траектория $\{u(\cdot), x(\cdot)\}$ допустима (т. е. $F_i[u(\cdot)] = 0$, $i = 1, \dots, m$; $u(t) \in U$) и не удовлетворяет принципу максимума. Тогда решение задачи (11)–(13) $\delta u^*(t) \neq 0$ и является улучшающей вариацией управления.

Доказательство. Негативная формулировка принципа максимума (5.2) утверждает, что при сделанных предположениях существует улучшающая вариация управления $\delta \tilde{a}(t) \neq 0$, т. е.:

$$1) \quad \int_0^T w_0(t) \delta \tilde{a}(t) dt < 0.$$

$$2) \quad u(t) + \delta \bar{u}(t) \in U.$$

$$3) \quad \int_0^T w_i(t) \delta \bar{u}(t) dt = 0, \quad i = 1, \dots, m.$$

Вариация $\delta \bar{u}(t)$ удовлетворяет условиям (12), (13). Используя еще стандартное предположение о выпуклости U , можно утверждать, что и $\mu \delta \bar{u}(\cdot)$ удовлетворяет (12), (13) при любом $0 \leq \mu \leq 1$. Тогда при достаточно малом μ

$$\int_0^T \left\{ w_0 \mu \delta \bar{u} + \frac{1}{2S} \mu^2 \|\delta \bar{u}\|^2 \right\} dt < 0,$$

и, следовательно,

$$\int_0^T \left\{ w_0 \delta u^* + \frac{1}{2S} \|\delta u^*\|^2 \right\} dt \leq \int_0^T \left\{ \mu w_0 \delta \bar{u} + \mu^2 \frac{1}{2S} \|\delta \bar{u}\|^2 \right\} dt < 0,$$

т. е. $\delta u^*(t) \neq 0$.

Теперь остался вопрос об алгоритме решения задачи (11)–(13). На первый взгляд и здесь нет никаких проблем, такой алгоритм, притом сходящийся со скоростью геометрической прогрессии, в сущности, давно известен. Это — алгоритм решения задачи строго выпуклого программирования (см. § 42). Основанием для его применения служит

Теорема 3. Пусть σ — выпуклое множество в пространстве измеримых функций $\delta u(t)$, определяемое условием $\delta u(t) \in \delta U(t)$ при всех t (мы считаем $\delta U(t)$ выпуклым при каждом $t \in [0, T]$). Пусть тело P в $(m+1)$ -мерном пространстве — образ σ в отображении

$$\xi^0 = \int_0^T \left\{ w_0(t) \delta u(t) + \frac{1}{2S} \|\delta u(t)\|^2 \right\} dt,$$

$$\xi^i = F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt, \quad i = 1, 2, \dots, m.$$

Тогда нижняя граница тела P есть граница строго выпуклого тела. Точный аналитический смысл этого утверждения состоит в следующем: для любого вектора $g = \{1, g_1, \dots, g_m\}$, $\min_{\xi \in P} (\xi, g)$ достигается в единственной точке. Другими словами, гиперплоскость G , ортогональная g , может быть опорной к нижней границе P только в одной точке.

Доказательство. Вычислим (ξ, g) через δu :

$$(\xi, g) = \int_0^T \left\{ w(t) \delta u(t) + \frac{1}{2S} \|\delta u(t)\|^2 \right\} dt + \sum_{i=1}^m g_i F_i[u(\cdot)], \quad (14)$$

где $w(t) = \sum_{i=0}^m g_i w_i(t)$. Минимизация по $\delta u(\cdot)$ интеграла сводится к минимизации подынтегрального выражения в каждой точке t независимо от остальных значений.

Итак, следует определить

$$\min_{\delta u \in \delta U} \left\{ w \delta u + \frac{1}{2S} \|\delta u\|^2 \right\}.$$

Но минимум выпуклой вниз функции $w \delta u + \frac{1}{2S} \|\delta u\|^2$ на выпуклом множестве δU достигается в единственной точке. В самом деле, предположив достижение минимума в двух разных точках, δu_1 и δu_2 , мы приходим к противоречию, так как в силу выпуклости δU отрезок $\lambda \delta u_1 + (1 - \lambda) \delta u_2$, $0 \leq \lambda \leq 1$, принадлежит δU , а в силу выпуклости вниз функции $w \delta u + \frac{1}{2S} \|\delta u\|^2$, ее значения на упомянутом отрезке меньше наибольшего из значений на концах отрезка. Таким образом определяется единственная функция $\delta u(t)$, минимизирующая (ξ, g) . (Правда, эту функцию можно произвольно изменить на множестве меры нуль, но это совершенно несущественно, точка ξ от этого не изменится.) Теорема доказана.

Обозначим через $\xi(g)$ точку минимума (ξ, g) при $\xi \in P$

$$(\xi(g), g) = \min_{\xi \in P} (\xi, g) \quad \text{или} \quad \xi(g) = \arg \min_{\xi \in P} (\xi, g). \quad (15)$$

Задача (11)–(13) может быть сформулирована следующим образом: найти в P точку (и ее прообраз в σ) вида λe с минимальным значением λ ($e = \{1, 0, \dots, 0\}$ — вектор в $(m+1)$ -мерном пространстве). Как известно (см. § 42), эта задача эквивалентна суперпозиции задач на безусловный экстремум:

$$\max_{(g, e)=1} \{ \min_{\xi \in P} (\xi, g) \}. \quad (16)$$

Функция $\Phi(g) \equiv \min_{\xi \in P} (\xi, g)$ в данном случае легко вычисляется, равно как и точка $\xi(g)$ при любом заданном векторе g . Максимизация $\Phi(g)$ может осуществляться методом подъема по градиенту: переход от вектора g к следующему вектору g' осуществляется по формуле (см. § 42)

$$g' = g + s \xi(g); \quad \xi(g) = \xi(g) - e(\xi(g), e),$$

а шаг процесса S определяется решением одномерной задачи

$$\max_{\xi} \min_{g \in P} (\xi, g + s\xi(g)).$$

Эта задача легко решается с любой необходимой точностью. Подробно на этом мы не останавливаемся, так как все эти элементы алгоритма входят в применявшийся в расчетах метод решения задачи (11)—(13) и описаны в § 49. Хотя сходимость алгоритма доказана, попытка использования его в практических расчетах оказалась неудачной из-за крайне медленной сходимости. Этот вычислительный эксперимент подробно освещен в § 49. Именно поэтому реализация метода проекции градиента потребовала создания специального алгоритма, работающего намного быстрее. Правда, он (см. § 49) дает не точное, а лишь приближенное решение задачи (11)—(13), но точное нам и не нужно, так как им определяется лишь вариация управления. Этот алгоритм, по существу, близок к используемому в методе последовательной линеаризации алгоритму решения задачи линейного программирования. Кстати, при $S = \infty$ задача (11)—(13) переходит в задачу линейного программирования, решение которой определяет вариацию управления в методе последовательной линеаризации (§§ 19, 21, 48).

Таким образом, задача (11)—(13) оказалась не столь уж простой, и хотя общая идея метода проекции градиента сформулирована очень давно, ее реализация применительно к достаточно общей задаче оптимального управления осуществлена, видимо, впервые в работе автора [96]. Решение конкретных задач описано в §§ 34, 37.

Однако для частных классов задач метод проекции градиента был предложен намного раньше. Эти частные классы выделяются тем, что задача проектирования, аналогичная задаче (11)—(13), оказывается более простой и решается привычными вычислительными методами. Можно выделить два класса таких задач.

Задачи классического типа. Так мы будем называть задачи, в постановке которых отсутствуют условия-неравенства, а именно: нет ограничений на управление $u(t) \in U$, которые обычно имеют вид системы неравенств $\varphi_j(u) \leq 0$, а дополнительные условия имеют вид $F_i[u(\cdot)] = 0$, $i = 1, \dots, m$ (исключаются условия вида $F_i \leq 0$). В этом случае проектирование сводится к определению

$$\min_{u(\cdot)} \int_0^T \left\{ w_0(t) \delta u(t) + \frac{1}{2S} \|\delta u(t)\|^2 \right\} dt, \quad (17)$$

при условиях

$$F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt = 0, \quad i = 1, 2, \dots, m. \quad (18)$$

Задача легко решается методом Лагранжа

$$\delta u(t) = -S w_0(t) + \sum_{i=1}^m \lambda_i w_i(t). \quad (19)$$

Множители λ_i находятся после подстановки (19) в условия (18) решением системы m линейных алгебраических уравнений.

Простейшие неклассические задачи. Так мы будем называть задачи, в которых есть условия-неравенства $u \in U$, но нет дополнительных условий $F_i = 0$, $i = 1, \dots, m$. В этом случае определение $\delta u(t)$ сводится к задаче отыскания

$$\min_{\delta u(\cdot)} \int_0^T \left\{ w_0 \delta u + \frac{1}{2S} \|\delta u\|^2 \right\} dt, \quad (20)$$

при условии

$$\delta u(t) \in \delta U(t) \quad \text{при всех } t. \quad (21)$$

Проектирование в функциональном пространстве здесь расщепляется на независимые (при разных t) задачи проектирования в конечномерном пространстве:

$$\min_{\delta u \in \delta U(t)} \left\{ w_0(t) \delta u + \frac{1}{2S} \|\delta u\|^2 \right\}. \quad (22)$$

В прикладных задачах геометрия области U обычно крайне проста, и определение $\delta u(t)$ из (22) можно считать элементарным. Наиболее частый в приложениях случай области U — прямоугольник в r -мерном пространстве (r — размерность u)

$$u \in U: \quad u^- \leq u(t) \leq u^+, \quad (23)$$

u^-, u^+ — заданные r -векторы; (23) следует понимать как систему r неравенств для r компонент u . В этом случае явное решение (22) легко выписывается (см., впрочем, § 45).

В а р и а н т ы. Метод проекции градиента настолько естествен, что предлагался многими авторами независимо друг от друга. Иногда эти предложения отличались только формой описания, иногда — деталями, не имеющими, видимо, принципиального значения. Полезно указать, хотя бы в общих чертах, эти внешне различные варианты метода.

Метод условного градиента (для задачи классического типа). Вариация управления $\delta u(\cdot)$ находится, как решение задачи

$$\min_{\delta u(\cdot)} \int_0^T w_0(t) \delta u(t) dt, \quad (24)$$

при условиях

$$F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt = 0, \quad i = 1, \dots, m, \quad (25)$$

$$\int_0^T \|\delta u(t)\|^2 dt = S^2. \quad (26)$$

Задача решается методом Лагранжа и приводит к конструкции типа (19) после решения той же самой системы m линейных алгебраических уравнений. Задачи (17), (18) и (24)–(26) эквивалентны и отличаются лишь способом введения шага S . Другими словами, между параметрами S , входящими в эти задачи, можно установить такое соответствие, при котором обе дают одну и ту же функцию $\delta u(t)$. Проверку этого факта предоставим читателю.

Метод минимальной поправки (для задач классического типа). Вариация управления ищется в форме $\delta u(t) = -Sw_0(t) + v(t)$, где $v(t)$ — поправка, компенсирующая вызванные вариацией $-Sw_0(t)$ нарушения дополнительных условий; эту поправку, естественно, следует взять минимальной. Таким образом, приходим к следующей задаче для определения $v(t)$: найти

$$\min_{v(\cdot)} \int_0^T \|v(t)\|^2 dt \quad (27)$$

при условиях

$$F_i[u(\cdot)] + \int_0^T w_i(t) [-Sw_0(t) + v(t)] dt = 0, \quad i = 1, \dots, m. \quad (28)$$

Задача (27), (28) эквивалентна задачам (17), (18) и (24)–(26), в указанном выше смысле.

Gradient-Restoration Algorithm [52], [53] (для задач классического типа). Разработанный в последние годы, метод основан на втором способе интерпретации задачи оптимального управления как общей задачи математического программирования и внешне существенно отличается от приведенных выше форм метода проекции градиента. Однако, кроме формального отличия, здесь есть и некоторое отличие по существу, влияние которого на алго-

ритм решения собственно задачи оптимального управления мы еще обсудим. Метод ориентирован в основном на задачи, в которых функционалы $F_t[u(\cdot), x(\cdot)]$, определяющие дополнительные условия, сформулированы в терминах значений $x(T)$. Таким образом, исходная вариационная задача предполагается имеющей следующий вид: найти

$$\min_{x, u} \int_0^T f^0(x, u) dt \quad (\min F_0[x(\cdot), u(\cdot)]) \quad (29)$$

при условиях

$$\dot{x} = f(x, u), \quad x(0) = X_0, \quad x(T) = X_1. \quad (30)$$

Такую форму имеет большое число прикладных задач. Впрочем, нетрудно обобщить метод и на более общий случай. В частности, не обязательно, чтобы при $t=0$ и при $t=T$ были заданы все значения компонент вектора x . Решение вариационной задачи и здесь представляет собой построение некоторой последовательности траекторий $\{u(\cdot), x(\cdot)\}$, причем условия $\dot{x} = f(x, u)$ предполагаются выполненными (с необходимой точностью) лишь в конце решения задачи. Стандартный шаг процесса состоит в переходе от некоторой траектории $\{u(\cdot), x(\cdot)\}$ к следующей $\{u(\cdot) + \delta u(\cdot), x(\cdot) + \delta x(\cdot)\}$; основной его элемент — это построение вариации $\{\delta u(\cdot), \delta x(\cdot)\}$. Варьируемая траектория $\{u(\cdot), x(\cdot)\}$ может либо удовлетворять (с заданной точностью) условиям (30), т. е., например,

$$\int_0^T \|\dot{x} - f(x, u)\|^2 dt \leq \varepsilon, \quad (31)$$

и тогда работает градиентный элемент алгоритма, либо условие (31) нарушено, и тогда работает восстанавливающий элемент алгоритма (Restoration Phase).

Рассмотрим первый случай. Итак, условие (31) выполнено, и вариация $\{\delta u(\cdot), \delta x(\cdot)\}$ имеет целью уменьшение значения минимизируемого функционала F_0 . Для этого задача линейризуется в окрестности траектории $\{u(\cdot), x(\cdot)\}$, что приводит к следующей задаче*) для $\{\delta u(\cdot), \delta x(\cdot)\}$: найти

$$\min_{\delta u, \delta x} \int_0^T \{f_x^0[t] \delta x(t) + f_u^0[t] \delta u(t)\} dt \quad (29^*)$$

*) Как обычно, мы используем обозначения

$$f[t] \equiv f[x(t), u(t)].$$

при условиях

$$\begin{aligned} \frac{d \delta x}{dt} - f_x[t] \delta x - f_u[t] \delta u &= - \left\{ \frac{dx}{dt} - f[t] \right\}, \\ \delta x(0) &= 0; \quad \delta x(T) = 0. \end{aligned} \quad (30^*)$$

Кроме того, добавляется условие, ограничивающее величину вариации некоторым малым числом S

$$\int_0^T \|\delta u(t)\|^2 dt \leq S^2. \quad (32)$$

Задача (29*), (30*), (32) может быть решена тем или иным способом. В работах, развивающих этот подход, выписывается необходимое условие экстремума (предполагается при этом, что невязки $\dot{x} - f[t]$ и $x(0) - X_0$, $x(T) - X_1$ пренебрежимо малы в соответствии с (31), поэтому соответствующие члены просто игнорируются). Это необходимое условие, как известно, имеет форму краевой задачи для системы $2n$ (n — размерность x) обыкновенных дифференциальных уравнений: к уравнению в вариациях (30*) с $2n$ краевыми условиями добавляется еще сопряженная система

$$-\frac{d\psi}{dt} - f_x^*[t] \psi = f_x^0[t], \quad (33)$$

и уравнение принципа максимума, связывающее в каждый момент времени значение $\delta u(t)$ с $\delta x(t)$ и $\psi(t)$ и позволяющее в данном случае однозначно выразить $\delta u(t)$ в виде линейной формы от $\delta x(t)$ и $\psi(t)$. Таким образом, дело сводится к решению краевой задачи для системы уравнений с $2n$ неизвестными $\delta x(t)$, $\psi(t)$, причем n условий ($\delta x(0)=0$) заданы при $t=0$, n — при $t=T$ ($\delta x(T)=0$).

Определение $\delta u(t)$, $\delta x(t)$ после нахождения (численно) фундаментальной системы частных решений сводится к системе n линейных алгебраических уравнений. Если при $t=0$ (или $t=T$) заданы не все значения компонент x , привлекаются соответствующие условия трансверсальности, и все делается точно так же, меняется только вид краевых условий. Мы ограничимся этим общим описанием, отправляя интересующихся к оригинальным работам. Для нас же важен основной вывод: задача (29*), (30*), (32) эквивалентна задаче (27), (28) и т. д., а с задачей (24)—(26) совпадает точно: при одинаковых значениях S в этих задачах будет получена одна и та же функция $\delta u(t)$. Однако после этого начинаются некоторые различия. Теперь мы получаем не только $\delta u(t)$, но и $\delta x(t)$, и новая траектория $\{u(t) + \delta u(t), x(t) + \delta x(t)\}$, вообще говоря, не удовлетворяет уравнению $\dot{x} = f(x, u)$, возникают не-

вязки в этом уравнении, имеющие величину $O(S^2)$, в то время как в краевом условии при $t=T$, если на варьируемой траектории не было невязок, то их не будет и на новой траектории (в силу $\delta x(T)=0$). В той реализации метода проекции градиента, которая будет использована в §§ 34, 37, так же как и в расчетах методом последовательной линеаризации, после определения $\delta u(t)$ находится новое управление $u(t) + \delta u(t)$, с этим управлением интегрируется система $\dot{x}=f$, $x(0)=X_0$ и возникают невязки $x(T) - X_1 = O(S^2)$ в условиях при $t=T$. Таким образом, пока выполнено условие (31), описываемый алгоритм дает практически ту же самую последовательность управлений (и фазовых траекторий), которую дал бы (при прочих равных условиях) любой из описанных выше вариантов метода проекции градиента. Основное отличие этих методов проявляется лишь на первой стадии решения задачи. В наших расчетах (см., например §§ 29—38) решение задачи начинается заданием некоторой функции $u(t)$, интегрируется система $\dot{x}=f(x, u)$, $\Gamma(x)=0$ (в данном случае $\Gamma(x)=0$ эквивалентно $x(0)-X_0=0$), получается траектория, не удовлетворяющая дополнительным условиям $F_i[u(\cdot)]=0$, $i=1, \dots, m$ (в данном случае $x(T)-X_1=0$), и далее первые итерации имеют целью, игнорируя изменение минимизируемого функционала, получить траекторию, удовлетворяющую всем условиям задачи. После того как такая траектория получена, начинается собственно процесс минимизации. В описываемом же алгоритме решение начинается заданием относительно произвольной пары функций $\{u(t), x(t)\}$, причем от $x(t)$ требуется выполнение условий как при $t=0$, так и при $t=T$. Обычно берется $x(t)=X_1 + (1-t/T)(X_0-X_1)$. Первые итерации и в этом случае имеют целью, не обращая внимания на F_0 , получить траекторию $\{u(\cdot), x(\cdot)\}$, удовлетворяющую и краевым условиям, и уравнению $\dot{x}=f(x, u)$. Это достигается вычислением поправок $\{\delta u, \delta x\}$, являющихся решением задачи

$$\min \int_0^T \|\delta u(t)\|^2 dt \quad (34)$$

при условиях

$$\begin{aligned} \frac{d\delta x}{dt} - f_x[t]\delta x - f_u[t]\delta u &= -\{\dot{x} - f[t]\}, \\ \delta x(0) &= 0; \quad \delta x(T) = 0. \end{aligned} \quad (35)$$

Если невязка $\dot{x} - f[t]$ велика (это обычно бывает в начале решения задачи), полученные поправки $\{\delta u(t), \delta x(t)\}$ не используются полностью, вводится однопараметрическое семейство траекторий

$\{u(t) + s \delta u(t), x(t) + s \delta x(t)\}$ и выбор s осуществляется с целью минимизировать норму невязки

$$\min_s \int_0^T \|\dot{x} + s \delta \dot{x} - f(x + s \delta x, u + s \delta u)\|^2 dt. \quad (36)$$

После того как получена траектория, удовлетворяющая условию (31), начинается процесс минимизации F_0 , в ходе которого постепенно накапливаются невязки в уравнении $\dot{x}=f$. Как только условие (31) оказывается нарушенным, снова решается задача (34), (35), но теперь уже, как правило, достаточно однократного вычисления поправок $\{\delta u(t), \delta x(t)\}$, и задача (36) не решается, а сразу берется исправленная траектория $\{u(t) + \delta u(t), x(t) + \delta x(t)\}$. В наших расчетах процесс минимизации F_0 объединен с процессом погашения невязок. Это можно сделать и в данном алгоритме, если при решении задачи (29*), (30*) не игнорировать правой части уравнения в вариациях (30*). Существенного значения для эффективности процесса решения вариационной задачи эта деталь, видимо, не имеет.

Метод проекции градиента и принцип максимума [40], [86] (для простейших неклассических задач). В задачах, не содержащих условий типа $F_i[u(\cdot)] = 0$, принцип максимума имеет особенно простую форму. Пусть имеется некоторая траектория $\{u(\cdot), x(\cdot)\}$ управляемой системы $\dot{x} = f(x, u)$, $x(0) = X_0$, которая исследуется на минимальность дифференцируемого по Фреше функционала

$$F_0[u(\cdot)] \equiv \int_0^T f^0[x(t), u(t)] dt + \Phi^0[x(T)] \quad (37)$$

(этот частный случай взят нами только в силу его распространенности как в приложениях, так и в теоретических работах). Тогда формулировка принципа максимума не содержит никаких неопределенных параметров: нужно найти решение задачи Коши, определенной на невозмущенной траектории,

$$-\frac{d\psi}{dt} - f_x^*[t] \psi = f_x^0[t], \quad \psi(T) = \Phi_x^0[x(T)], \quad (38)$$

вычислить функцию

$$H[x(t), \psi(t), u(t)] \equiv -f^0[x(t), u(t)] - (\psi(t), f[x(t), \psi(t)]) \quad (39)$$

и проверить условие

$$H[x(t), \psi(t), u(t)] = \max_{u \in U} H[x(t), \psi(t), u]. \quad (40)$$

Пусть это условие не выполнено, и траектория, таким образом, не оптимальна. Естественно возникает мысль решить задачу методом итераций, совпадающим по алгоритмической схеме с известным методом Пикара: определим новое управление $u^*(t)$ из уравнения

$$H[x(t), \psi(t), u^*(t)] = \max_{u \in U} H[x(t), \psi(t), u]. \quad (41)$$

С этим управлением $u^*(t)$ решим задачу Коши: $\dot{x} = f(x, u^*)$, $x(0) = X_0$, снова вычислим ψ из (38) и т. д. Подобные методы сходятся только при счастливом стечении обстоятельств. Расходимость (точнее, отсутствие сходимости) этого процесса быстро обнаружилась, была понятна и причина: при не очень хорошем начальном приближении изменение управления ($\delta u = u^* - u$), слишком велико, величинами $O(\|\delta u\|^2)$ пренебрегать нельзя (не следует забывать, что вывод уравнения (40) основан на теории возмущений первого порядка). Нужно было исправить метод, сделав $\delta u(t)$, при необходимости, малым. Такое усовершенствование было предложено в [86]. Именно, после определения $u^*(t)$ из (41) образуется однопараметрическое семейство управлений

$$u(t, s) = u(t) + s[u^*(t) - u(t)]. \quad (42)$$

Значение шага s может определяться, например, решением одномерной задачи: найти

$$\min_s F_0[u(\cdot, s)]. \quad (43)$$

Определение s (приближенное, разумеется) «стоит» нескольких интегрирований системы $\dot{x} = f$ и вычислений F_0 по формуле (37).

Другой вариант введения в процесс малого параметра использован в [19] (см. также § 37). В этом случае после определения $u^*(t)$ из (41) на интервале $[0, T]$ выделяется множество M_s условием

$$t \in M_s: \quad H[x(t), \psi(t), u^*(t)] \geq \max_t H[x(t), \psi(t), u^*(t)] - s, \quad (44)$$

после чего новое управление $u(t, s)$ вычисляется следующим образом:

$$u(t, s) = \begin{cases} u(t) & \text{при } t \notin M_s, \\ u^*(t) & \text{при } t \in M_s, \end{cases} \quad (45)$$

а параметр $s > 0$ определяется, например, той же задачей (43). Метод проекции градиента в данной ситуации привел бы к следующим вычислениям: после определения $\psi(t)$ из (38) вычисляются градиент F_0 и его проекция

$$\begin{aligned} w_0(t) &= f_u^0[t] + f_u^*[t]\psi(t), \\ u(t, s) &= P_U\{u(t) - sw_0(t)\}. \end{aligned} \quad (46)$$

Методы, основанные на конструкциях (42) и (45), кажутся более привлекательными, так как они непосредственно связаны с основным теоретическим результатом — принципом максимума. Однако в большинстве случаев это лишь видимость преимущества, хотя и считать конструкцию (46) существенно лучшей тоже нет оснований. В самом деле, сравним (46) с (42). Сходимость метода самым серьезным образом связана с тем, что на получаемой последовательности управлений значение функционала F_0 монотонно понижается. То, что (46) приводит к понижению F_0 , очевидно. Проверим это для конструкции (42), вычислив производную

$$\begin{aligned} \frac{dF_0[u(\cdot, s)]}{ds} \Big|_{s=0} &= \left(\frac{\partial F_0[u(\cdot)]}{\partial u(\cdot)}, u^*(\cdot) - u(\cdot) \right) = \\ &= \int_0^T (w_0(t), u^*(t) - u(t)) dt. \end{aligned} \quad (47)$$

Теперь заметим, что

$$w_0(t) = f_u^0[t] + f_u^*[t] \psi(t) = -H_u[x(t), \psi(t), u(t)].$$

Таким образом,

$$\frac{dF_0}{ds} \Big|_{s=0} = - \int_0^T (H_u[x(t), \psi(t), u(t)], u^*(t) - u(t)) dt. \quad (48)$$

В общем случае, из $H[x, \psi, u^*] > H[x, \psi, u]$ не следует $(H_u[x, \psi, u], u^* - u) > 0$, понижение F_0 с ростом s не гарантируется (отрицательные значения s могут быть запрещены условием $u(t, s) \in U$). Однако в очень распространенной ситуации, при линейной зависимости всех функций $f(x, u)$ от u , $(H_u, u^* - u) > 0$, и метод оказывается сходящимся. В общем случае сходимости может не быть при сколь угодно хорошем начальном приближении.

Конструкция (45) с точки зрения сходимости более естественна и логична: ведь глобальный характер (по $u \in U$) уравнения принципа максимума связан с использованием конечных вариаций на множествах малой меры, и конструкция (45) в отличие от (42), это учитывает. В § 6 была получена формула для приращения функционала, вызванного конечным изменением управления на множестве M_μ малой меры $\mu_\mu = \text{mes } M_\mu$:

$$\begin{aligned} F_0[\tilde{u}(\cdot, s)] - F_0[u(\cdot)] &= \int_0^T \{f^0(x, \tilde{u}) - f^0(x, u) + \\ &+ (\psi, f(x, \tilde{u}) - f(x, u))\} dt + O(\mu_\mu^2). \end{aligned} \quad (49)$$

Эта формула показывает, что при достаточно малой величине $\mu_\mu = \text{mes } M_\mu$ переход от $u(\cdot)$ к $\tilde{u}(\cdot, s)$ сопровождается понижением

значения F_0 (если, конечно, траектория $\{u(\cdot), x(\cdot)\}$ не удовлетворяет принципу максимума).

Метод проекции градиента и скользящие режимы. Следует особо отметить те задачи, в которых конструкция (45) будет иметь значительное преимущество перед методом проекции градиента в форме (46), (43). Это — задачи, где оптимальная траектория содержит участок так называемого «скользящего режима» (см. § 23). В этом случае могут существовать неоптимальные траектории, на которых конструкция (46) при не слишком больших s дает функцию $u(t, s) = u(t)$; такая траектория оказывается тупиковой для методов (46), (43). В то же время конструкция (45) приводит к ненулевой вариации управления: $\tilde{u}(t, s) \neq u(t)$. Пример, рассмотренный в § 23, показывает, что эта возможность действительно реализуется при численном решении подобных задач, причем множество тупиковых для локального варианта проекции градиента (46) траекторий достаточно мощно и содержит траектории, далекие от оптимальной. Тем не менее, в дальнейшем мы будем иметь дело именно с локальным вариантом. Это связано с тем, что среди известных автору прикладных задач, решавшихся приближенными методами, нет задач, содержащих скользящие режимы. Более того, в монографиях [39], [102], посвященных преимущественно обобщению теории вариационных задач, охватывающему и скользящие режимы (что, разумеется, приводит к серьезному усложнению аналитического аппарата теории), подобных примеров тоже нет! Речь, разумеется, идет о примерах задач, естественно возникших в приложениях, а не специально сконструированных с целью иллюстрации тех или иных возможных осложнений. С этой точки зрения те предостережения, которые делает инженерам и физикам автор [102] в связи с «наивным» использованием результатов классического вариационного исчисления, представляются преувеличенными. Разумеется, практика решения вариационных задач может расширяться, и задачи со скользящими режимами станут обычным, «инженерным» явлением. В этом случае изменится и отношение к соответствующему разделу в теории, и в вычислительные методы будут внесены необходимые коррективы.

Задачи с параметром и одним дополнительным условием. Задачи без дополнительных условий привлекательны прежде всего тем, что направление варьирования управления (проекция градиента) находится очень просто, а выбор шага S осуществляется не очень сложным и вполне объективным способом. Поэтому, если есть возможность естественного сведения исходной задачи к задаче без дополнительных условий, ей следует воспользоваться. Есть класс задач, не очень широкий, впрочем, в котором можно избавиться от дополнительных условий и получить простейшую задачу неклассического типа. Это

задачи, содержащие, например, одно условие и один управляющий параметр, в качестве которого часто фигурирует время управления T . Рассмотрим задачу

$$\min_{u, T} F_0[u(\cdot), T] \text{ при условиях } \dot{x} = f(x, u), x(0) = X_0, \quad (50)$$

$$F_1[u(\cdot), T] = 0.$$

В этом случае условие $F_1[u(\cdot), T] = 0$ используется в качестве признака окончания интегрирования системы $\dot{x} = f$, $x(0) = X_0$, и всегда выполнено на очередной варьируемой траектории $\{u(\cdot), x(\cdot)\}$. Оба функционала предполагаем дифференцируемыми, т. е. могут быть вычислены функции $w_i(t)$ и числа a_i , $i = 0, 1$ (см. § 7), и в первом порядке

$$F_i[u(\cdot) + \delta u(\cdot), T + \delta T] = F_i[u(\cdot), T] + \int_0^T w_i \delta u dt + a_i \delta T.$$

Условие $F_i[u(\cdot) + \delta u(\cdot), T + \delta T] = 0$ используется для определения

$$\delta T = -\frac{1}{a_1} \int_0^T w_1(t) \delta u(t) dt.$$

Подставляя это значение в выражение для δF_0 , получим

$$\delta F_0[\delta u(\cdot)] = \int_0^T \left(w_0 + \frac{a_0}{a_1} w_1, \delta u \right) dt = \int_0^T \tilde{w}_0(t) \delta u(t) dt.$$

По существу, $\tilde{w}_0(t)$ есть проекция градиента на линейное подпространство, касательное к многообразию, выделенному условием $F_1[u(\cdot), T] = 0$.

Часто в приложениях встречаются условия вида $F_1[u(\cdot), T] \equiv \Phi^1[x(T), T]$. В этом случае сразу вычисляем

$$\frac{\partial F_1}{\partial T} \delta T = \Phi_T^1 \delta T + \Phi_x^1 \frac{dx(T)}{dT} \delta T = (\Phi_T^1 + \Phi_x^1 f)_{x(T)} \delta T = a_1 \delta T.$$

Предоставляем читателю убедиться, что можно обойтись лишь одним интегрированием сопряженной системы и сразу вычислить $\tilde{w}_0(t)$, минуя вычисление и w_0 , и w_1 . Разумеется, следует предположить, что $a_1 \neq 0$. В противном случае условие $F_1[u(\cdot), T] = 0$, вообще говоря, нельзя использовать как признак окончания интегрирования. Заметим, что число дополнительных условий типа $F_i[u(\cdot)] = 0$ является одним из главных факторов, определяющих вычислительную трудность задачи: чем больше таких условий, тем труднее задача. Поэтому желательно, если есть такая возможность, это число сокращать. Если в задаче с m условиями $F_i = 0$ есть еще по крайней мере m управляющих параметров

$\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$, причем на эти параметры не наложено ограничений-неравенств, то несложно провести аналогичное проектирование. Обозначив здесь $F = \{F_1, F_2, \dots, F_m\}$, напомним формулу

$$\delta F[\delta u(\cdot), \delta \alpha] = \int_0^T W(t) \delta u(t) dt + A \delta \alpha,$$

где $W(t)$ — матрица $r \rightarrow m$ (r — размерность u), A — матрица $m \rightarrow m$. Вычисление W и A , вообще говоря, требует m -кратного интегрирования сопряженного уравнения. При определении $\delta u(\cdot)$ и $\delta \alpha$ должны быть выполнены условия

$$F[u(\cdot), \alpha] + \int_0^T W(t) \delta u(t) dt + A \delta \alpha = 0$$

и, если $\det A \neq 0$, можно провести «проектирование» — исключение $\delta \alpha$:

$$\delta \alpha = -A^{-1}F - \int_0^T A^{-1}W(t) \delta u(t) dt;$$

после чего выражение для δF_0 приобретает вид

$$\delta F_0 = \int_0^T w_0 \delta u dt + (a_0, \delta \alpha) = -(a_0, A^{-1}F) + \int_0^T (w_0 - W^* A^{*-1} a_0, \delta u) dt.$$

Не следует, однако, упускать из виду, что основное достоинство задач без дополнительных условий состоит в возможности использовать для выбора шага s задачу (43), не заботясь о том, какой, большой или малой, окажется при этом величина s . То же самое относится и к задаче с одним дополнительным условием и свободным временем. В этом случае мы все время имеем дело с траекториями, лежащими на многообразии $F_1[u(\cdot), T] = 0$, и можем не заботиться о величине шага s и о влиянии неучтенных при выборе вариации управления величин $O(\|\delta u\|^2)$. Это же относится еще к одному классу задач, в которых система уравнений $\dot{x} = f(x, u)$ линейна по x и u , а дополнительные условия поставлены также в терминах линейных как по x , так и по u функционалов (часто, например, такие условия имеют вид $x(T) = X_1$).

Метод проекции градиента в задачах с конечными связями. Рассмотрим задачу, в которой, кроме перечисленных в (7) условий, добавлена еще конечная связь между $x(t)$ и $u(t)$ вида

$$G[x(t), u(t)] = 0 \text{ при } t \in [t', t''] \subset [0, T]. \quad (51)$$

Будем для простоты считать $G(x, u)$ скалярной гладкой функцией. Наиболее простым является случай, когда $G(x, u)$ явно зависит хотя бы от одной компоненты управления, т. е. $G_u(x, u) \neq 0$ при всех $t \in [t', t'']$. В принципе в этом случае можно на интервале $[t', t'']$ выразить из (51) одну из компонент $u(t)$ через $x(t)$ и остальные компоненты $u(t)$ и перейти на $[t', t'']$ к системе уравнений с другой правой частью. Вычисление функциональных производных в этом случае достаточно просто (см. § 7). Более сложен случай, когда ограничение имеет вид

$$G[x(t)] = 0, \quad t \in [t', t'']. \quad (52)$$

Формируя линеаризованную задачу (29*), (30*), (32), можно дополнить ее и линеаризацией (52):

$$G[x(t)] + G_x[t] \delta x(t) = 0, \quad t \in [t', t'']. \quad (53)$$

Однако с таким условием задача определения $\delta x(t)$ и $\delta u(t)$ становится слишком сложной. Поэтому обычно предпочитают следующий путь: вводится функция

$$G^{(1)}(x, u) \equiv G_x(x) \dot{x}(t) = G_x(x) f(x, u). \quad (54)$$

Пусть эта функция уже содержит u явно, т. е. $G_u^{(1)}[x(t), u(t)] \neq 0$. Тогда вместо условия (52) используется

$$G^{(1)}[x(t), u(t)] = 0, \quad (55)$$

дополненное еще условием входа

$$G[x(t')] = 0 \quad (F[u(\cdot)] \equiv G[x(t')] = 0). \quad (56)$$

Последнее сформулировано в терминах дифференцируемого функционала и включается в стандартную группу дополнительных условий. Если $G_u^{(1)} \equiv 0$, т. е. (54) приводит к функции $G^{(1)}(x)$, следует ввести функцию

$$G^{(2)}(x, u) \equiv G_x^{(1)}(x) f(x, u) \quad (57)$$

и еще одно условие входа

$$G^{(1)}[x(t')] = 0. \quad (58)$$

Если и $G^{(2)}$ не содержит явно u , переходим к $G^{(3)}$, добавляя еще одно условие $G^{(2)}[x(t')] = 0$, и т. д. Получив в конце концов условие

$$G^{(k)}[x(t), u(t)] = 0 \quad (59)$$

и последовательность дополнительных условий

$$G[x(t')] = G^{(1)}[x(t')] = \dots = G^{(k-1)}[x(t')] = 0, \quad (60)$$

можно поступить двояко. В вычислительной схеме Gradient-Restoration Algorithm дополняют задачу (29*), (30*), (32) еще условиями

$$\begin{aligned} G^{(i)}[x(t')] + G_x^{(i)}\delta x(t') &= 0, \quad i=0, 1, \dots, k-1, \\ G^{(k)}[t] + G_x^{(k)}[t]\delta x(t) + G_u^{(k)}[t]\delta u(t) &= 0, \quad t \in [t', t'']. \end{aligned} \quad (61)$$

Заметим, что можно использовать (61) в однородной форме, пренебрегая невязками $G^{(i)}[x(t')]$, $i=0, 1, \dots, k$ до тех пор, пока не накопится некоторая суммарная невязка (соотношение (31) должно быть очевидным образом обобщено), после чего включается «восстанавливающая» форма алгоритма. Можно использовать, не вводя функций $G^{(i)}$, условие для $\delta x(t)$ в виде (55), что, конечно, осложняет задачу определения поправок $\{\delta u(t), \delta x(t)\}$. Вычислительная схема решения вариационных задач, используемая автором, предполагает использование условных функциональных производных, определенных уже с учетом соотношений типа (61) (см. § 21). Вычисление этих производных есть не что иное, как проектирование производных всех входящих в постановку задачи функционалов на подпространство, касательное к многообразию, определяемому условием $G[x(t), u(t)]=0$. Не нужно думать, что такой способ действий приведет к существенно другим функциям $\{\delta x(t), \delta u(t)\}$. Просто в этом случае операция проектирования разлагается на последовательность двух операций: сначала все проектируется на подпространство, касательное к $G[x, u]=0$, а затем, уже в этом подпространстве, осуществляется проектирование на пересечение подпространств, касательных к многообразиям, определяемым остальными условиями задачи. Результат от этого не меняется. В своих расчетах автор обычно не использовал ни одной из перечисленных выше форм проектирования. Дело в том, что в большинстве случаев в прикладных задачах появляется не условие в виде равенства (51), а условие в виде неравенства

$$G[x(t), u(t)] \leq 0 \quad \text{при} \quad t \in [t', t'']. \quad (51^*)$$

Оно может быть записано в стандартной форме

$$F[u(\cdot)] \leq 0, \quad \text{где} \quad F[u(\cdot)] \equiv \max_{[t', t'']} G[x(t), u(t)], \quad (62)$$

с функционалом F , не имеющим, вообще говоря, производной Фреше. Решение задач с подобными функционалами подробно описано в §§ 21, 34—38.

Проектирование на множество в функциональном пространстве, выделенное неравенством (51*), является очень сложной операцией, так как «касательное» к нему многообразие, описываемое неравенством

$$G_x[t]\delta x(t) + G_u[t]\delta u(t) \leq 0,$$

не является линейным подпространством: это есть выпуклый конус, и проектирование нельзя осуществить операциями линейной алгебры.

П а л л и а т и в ы (метод проекции градиента в общем случае). Выше было показано, что проектирование градиента осуществляется достаточно просто (правда, в линеаризованной постановке, приводящей к проектированию на линейное подпространство) в двух случаях: либо при отсутствии дополнительных условий ($F_i=0$), либо при отсутствии геометрического ограничения на значения $u(t)$ ($u \in U$). Однако большая часть прикладных задач оптимального управления содержит оба сорта условий, а в этом случае проектирование выполняется решением задачи квадратического программирования. К сожалению, идеи и алгоритмы, относящиеся к линейному и нелинейному программированию, мало известны среди специалистов по прикладной механике, которые особенно часто сталкиваются с необходимостью решения задач оптимального управления достаточно общего вида. Именно в этой среде были созданы многочисленные приемы, имеющие целью сформулировать общую задачу как задачу классического типа, либо как простейшую неклассическую задачу. Мы рассмотрим наиболее типичные из этих приемов. Их следует отнести к разряду паллиативов, так как они не снимают трудностей численного решения, а лишь отодвигают их, так сказать, в глубь проблемы. Создание алгоритма приближенного решения задачи оптимального управления можно условно разбить на два этапа:

- 1) предложение некоторой конструкции;
- 2) реализация этой конструкции в виде программы на ЭВМ и испытание метода решением различных реальных и модельных задач.

Приемы, о которых сейчас пойдет речь, облегчают первый этап, перенося трудности на второй:

Метод штрафных функций позволяет любую самую общую задачу оптимального управления свести (приближенно, но с любой необходимой степенью точности) и к простейшей неклассической, и к задаче классического типа, и, наконец, к задаче, в которой нет ни условий типа $F_i[u(\cdot)]=0$, ни геометрических ограничений $u(t) \in U$. Однако это достигается ценой введения в задачу больших параметров, что в свою очередь приводит к функционалу с соответственно малой областью точности линейного приближения. Минимизация подобных функционалов оказывается крайне сложной, а полученные результаты — не очень надежными. Этим мы здесь и ограничимся, так как методу штрафных функций посвящен отдельный параграф (§ 50).

Замена управления, позволяющая «раскрыть» область U , т. е. из ограниченной замкнутой области U в условии $u(t) \in U$

сделать «открытую» и, по существу, исключить условие $u(t) \in U$ из постановки задачи. Тем самым задача становится задачей классического типа.

Пусть условие $u(t) \in U$ фактически задается неравенством $|u(t)| \leq 1$ (для каждой компоненты u). Тогда, вводя вместо $u(t)$ новую управляющую функцию $v(t)$ и полагая (покомпонентно)

$$u(t) = \sin v(t), \quad (63)$$

мы можем никаких условий на $v(t)$ не накладывать. Такая замена часто рекомендуется для практического использования, при этом недостатком этого приема считают слишком узкую область применения — только к условиям вида $|u| \leq 1$. Если бы дело было в этом, беда была бы не велика. Ведь именно такие ограничения на u появляются чаще, чем какие-нибудь другие, да и в более сложных случаях можно было бы придумать что-либо подобное.

Все было бы очень хорошо, если бы нашей целью было только избавиться от условий типа $u \in U$. К сожалению, этим дело не кончается: ведь этот результат нужен нам для того, чтобы использовать простую процедуру проектирования в задачах классического типа. Функциональные производные $w_i(t) = \frac{\partial F_i[u(\cdot)]}{\partial u(\cdot)}$ элементарно пересчитываются в производные по $v(\cdot)$:

$$\frac{\partial F_i[u(\cdot)]}{\partial v(\cdot)} = \frac{\partial F_i}{\partial u(\cdot)} \cos v(t) = w_i(t) \cos v(t). \quad (64)$$

Проектирование градиента приводит к конструкции

$$\delta v(t) = -S w_0(t) \cos v(t) + \sum_{i=1}^m \lambda_i w_i(t) \cos v(t), \quad (65)$$

и мы сталкиваемся со следующим неприятным явлением: если на некотором интервале $[t_1, t_2]$ управление u вышло на границу ($u=1$ или -1), то на этом интервале $\cos v(t)=0$ и, следовательно, $\delta v(t)=0$, т. е. управление u «прилипло» к границе, хотя это может и не соответствовать существу дела. Замена (63) порождает в пространстве функций $v(\cdot)$ мощное множество «псевдостационарных» точек, стационарных для управления v , но не являющихся стационарными для управления u .

Преобразование Валентайна (введение дополнительного управления) позволяет избавиться от неравенств в постановке задачи, заменив их эквивалентными равенствами. Пусть в задаче фигурирует условие

$$\varphi[x(t), u(t)] \leq 0, \quad (66)$$

где φ будем считать, для простоты, скалярной функцией. Тогда, вводя дополнительную компоненту управления $v(t)$, запишем условие (66) в виде равенства

$$\varphi[x(t), u(t)] + v^2(t) = 0. \quad (67)$$

Однако и здесь все хорошо до тех пор, пока мы не приступаем к построению вариаций $\delta x(t)$, $\delta u(t)$ методом проектирования градиента. В самом деле, при построении вариации $\{\delta x, \delta u\}$ условие (63) учитывается в форме

$$\varphi_x \delta x(t) + \varphi_u \delta u(t) + 2v(t) \delta v(t) = 0, \quad (68)$$

и если на некотором интервале $[t_1, t_2]$ на варьируемой траектории $\{u(t), x(t)\}$ в условии (66) реализуется равенство, т. е. в (67) $v(t) = 0$, то и (68) превращается в $\delta\varphi = \varphi_x \delta x + \varphi_u \delta u = 0$, т. е. мы сталкиваемся с тем же самым явлением «прилипания» управления к границе.

Итерационный метод работы с неравенствами, как с равенствами, был предложен в [51]. Суть дела поясним на самом простом примере. Пусть в задаче есть одно условие-неравенство $\varphi[u(t)] \leq 0$. Тогда процесс построения вариации $\delta u(t)$ начинается с того, что все неравенства игнорируются и проекция градиента вычисляется классическими методами. Найденная вариация $\delta u(t)$ может привести к нарушению условия $\varphi \leq 0$. Пусть на некотором интервале $[t_1, t_2]$ окажется $\varphi[u(t) + \delta u(t)] > 0$. Тогда на этом интервале условие $\varphi \leq 0$ заменяется условием-равенством $\varphi[u(t)] = 0$, и находится новая вариация $\delta u(t)$ в задаче, поставленной только в терминах равенств, снова проверяется условие $\varphi \leq 0$ и т. д. Однако этот эрзац операции проектирования теоретически несостоятелен: в простых ситуациях он может привести к $\delta u(t) \equiv 0$, хотя варьируется траектория очевидным образом неоптимальная, и правильное проектирование градиента привело бы, конечно, к $\delta u(t) \neq 0$.

Рассмотрим тот же пример, который был использован в § 16 для демонстрации аналогичного дефекта метода локальных вариаций. Здесь нам не очень важны детали, важно следующее:

- 1) управление ограничено условием $0 \leq u(t) \leq 1$;
- 2) варьируемое управление $u(t)$ имеет вид

$$u(t) = \begin{cases} 1, & 0 \leq t < t_1, \\ 0, & t_1 \leq t \leq T; \end{cases}$$

3) функция $-w_0(t)$ (w_0 — градиент минимизируемого функционала) изображена на рис. 15.

Кроме того, одним из условий проектирования является соотношение

$$\int_0^T \delta u(t) dt = 0. \quad (69)$$

Следуя [51], игнорируем условия-неравенства $0 \leq u \leq 1$ и находим проекцию $-w_0$ только на условие (69): $\delta u^{(1)}(t) = -w_0(t) + \lambda$, где λ находится из соотношения

$$\begin{aligned} \int_0^T \delta u^{(1)}(t) dt &= \\ &= - \int_0^T w_0(t) dt + \lambda T = 0. \end{aligned}$$

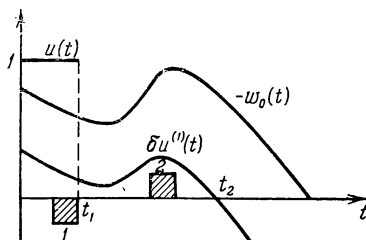


Рис. 15.

Функция $\delta u^{(1)}(t)$ изображена на том же рисунке. Видно, что условие $0 \leq u(t) + \delta u(t) \leq 1$ нарушено на интервалах $[0, t_1]$ и $(t_2, T]$. Следовательно, в дальнейшем при построении $\delta u(t)$ на этих интервалах полагается $\delta u(t) = 0$. На оставшемся интервале (t_1, t_2) возможна только вариация $\delta u(t) \geq 0$, что совместно с условием (69) дает $\delta u(t) \equiv 0$. Между тем элементарно находится вариация $\delta u(t)$, состоящая из двух финитных элементов 1 и 2 (см. рис. 15), для которой

$$\int_0^T \delta u(t) dt = 0 \quad \text{и} \quad \int_0^T w_0(t) \delta u(t) < 0.$$

Итак, приемы 2—4 в общем случае не позволяют строить минимизирующую последовательность управлений. Некоторые заведомо не оптимальные ситуации являются тупиковыми для этих методов, поэтому доказать их сходимость, не вводя каких-то существенных усовершенствований, в принципе нельзя. Однако не следует думать, что используя такие приемы, нельзя получить достаточно точные приближенные решения. Ведь тупиковыми для этих методов построения $\delta u(t)$ являются далеко не все неоптимальные траектории $\{u(\cdot), x(\cdot)\}$. Не исключена возможность добраться до оптимальной траектории, так и не встретив по пути тупиковой. В частности, успех возможен, если процесс поиска происходит, так сказать, «прямолинейно»: если на каком-то этапе поиска в какой-то точке t в условии-неравенстве типа (66) реали-

зовался знак равенства, то это так и должно быть на оптимальной траектории. Неприятности будут в том случае, если в дальнейшем потребуются снова перейти к строгому неравенству. В примерах конкретных расчетов, которые встретятся в этой книге, читатель без труда заметит многочисленные примеры именно такого «прямолинейного» хода решения задач с неравенствами. Однако не следует недооценивать и частоту тех случаев, когда осуществляется «сход» с ограничения (знак $=$ в условии сменяется знаком $<$). Но подобные ситуации являются лишь короткими эпизодами в эволюции управления от исходного к оптимальному. Поэтому они не так заметны в тех неполных данных, которые иллюстрируют процесс решения задачи.

§ 19. Метод последовательной линеаризации

В этом параграфе и в §§ 20, 21 будет подробно описан метод решения весьма общего класса вариационных задач, разработанный и применявшийся автором с 1962 г. Первые публикации, содержащие достаточно полное описание метода и примеры решения прикладных задач, относятся к 1964 г. В дальнейшем, по мере накопления опыта практической работы, отдельные элементы вычислительной технологии были уточнены и улучшены.

Описываемый ниже метод является типичным методом спуска в пространстве управлений (методом построения минимизирующей последовательности управлений). Ниже будут очень подробно описаны не только принципиальная схема метода, но и детали вычислительной технологии. Второстепенные на первый взгляд, они требуют достаточно ответственного и квалифицированного решения. От того, насколько удачно решены эти вопросы, часто самым существенным образом зависит эффективность метода в целом. Есть и другая причина, побуждающая нас к столь подробному изложению. Дело в том, что при описании других методов спуска в пространстве управлений мы ограничились изложением лишь их основной конструктивной идеи. Практическая реализация этих методов неизбежно потребует решения целого ряда вопросов, которые мы условно относим к вычислительной технологии. Мы не излагали соответствующих рекомендаций, во-первых, потому, что они часто отсутствуют и в оригинальных работах, а во-вторых, потому, что они аналогичны тем, которые подробно будут описаны в §§ 20, 21.

П р и н ц и п и а л ь н а я с х е м а м е т о д а. Основным элементом построения минимизирующей последовательности управлений — это конструкция малого конечного возмущения управления $\delta u(t)$, с помощью которого осуществляется переход от данного управления $u(t)$ к улучшенному $u(t) + \delta u(t)$.

Изложение будет проведено на примере относительно простой задачи — найти управление $u(\cdot)$, минимизирующее значение функционала F_0 :

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (1)$$

на траектории управляемой системы

$$\frac{dx}{dt} = f(x, u); \quad \Gamma(x) = 0; \quad 0 \leq t \leq T \quad (2)$$

при дополнительных условиях

$$F_i[u(\cdot)] = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \quad (3)$$

и геометрическом ограничении

$$u(t) \in U \text{ при всех } t. \quad (4)$$

Все входящие в постановку задачи функционалы будем считать дифференцируемыми по Фреше; таким образом, мы пока не рассматриваем задач с функционалами типа

$$\max_t \Phi[x(t)], \quad \max_t \Phi[x(t), u(t)], \quad \int_0^T |\Phi(x, u)| dt. \quad (5)$$

Методы решения задач с такими функционалами будут описаны отдельно. Большая часть обобщений задачи, описанных в § 7, по существу, не сказывается на самом методе, влияя лишь на технику вычисления функциональных производных.

Итак, пусть известно некоторое управление $u(t)$, которое мы будем называть *невозмущенным*; выполнение дополнительных условий (3) не предполагается, геометрическое же ограничение $u(t) \in U$ будем считать выполненным; задание разумного исходного управления такого сорта в практических задачах затруднений не вызывает.

1. С данной функцией $u(t)$ решается краевая задача (2) и вычисляются значения функционалов $F_i[u(\cdot)]$.

2. В окрестности невозмущенной траектории $\{u(\cdot), x(\cdot)\}$ задача линеаризуется. Линеаризация задачи включает в себя два основных момента:

1) вычисление функциональных производных

$$w_i(t) = \frac{\partial F_i[u(\cdot)]}{\partial u(\cdot)}, \quad i = 0, 1, \dots, m; \quad (6)$$

2) построение некоторой малой окрестности $\delta U(t)$ невозмущенного управления.

Если первый вопрос решается однозначно использованием стандартной техники дифференцирования функционалов, то второй,

имеющий важное технологическое значение, однозначного решения не имеет. При построении $\delta U(t)$ должны быть учтены следующие естественные требования:

- а) из $\delta u(t) \in \delta U(t)$ должно следовать $u(t) + \delta u(t) \in U$;
- б) $\delta U(t)$ должна быть достаточно малой окрестностью, чтобы формулы первого порядка

$$\delta F_i[\delta u(\cdot)] = \int_0^T w_i(t) \delta u(t) dt$$

с достаточной точностью описывали точные приращения функционалов

$$\Delta F_i \equiv F_i[u(\cdot) + \delta u(\cdot)] - F_i[u(\cdot)];$$

- в) $\delta U(t)$ должна быть достаточно большой окрестностью, чтобы процесс перехода $u(\cdot) \rightarrow u(\cdot) + \delta u(\cdot)$ был не слишком медленным;

д) конструкция окрестности $\delta U(t)$ должна по возможности облегчить задачу нахождения $\delta u(t)$;

- е) $\delta U(t)$ должна быть полной окрестностью управления $u(\cdot)$ в смысле следующего определения.

Окрестность $\delta U(t)$ будем называть *полной*, если любая вариация управления $v(t)$, задающая возможное направление смещения управления (т. е. $u(t) + sv(t) \in U$ при всех t и $0 \leq s \leq s^*$) содержится в окрестности $\delta U(t)$: при некотором $s^{**} > 0$

$$sv(t) \in \delta U(t) \text{ при всех } t \text{ и } 0 \leq s \leq s^{**}.$$

Это свойство достаточно важно: используя неполные окрестности, можно превратить неоптимальную траекторию в оптимальную (относительно неполного множества вариаций управления) или, по крайней мере, замедлить процесс минимизации, так как метод будет использовать не все возможные пути движения управления к искомому оптимальному.

3. Формулируется и решается задача определения вариации $\delta u(\cdot)$, являющаяся линеаризацией решаемой задачи в окрестности невозмущенной траектории $\{u(\cdot), x(\cdot)\}$:

найти $\delta u(\cdot)$ из условий

$$\min_{\delta u(\cdot)} \delta F_0[\delta u(\cdot)] = \min_{\delta u(\cdot)} \int_0^T w_0(t) \delta u(t) dt, \quad (7)$$

$$F_i[u(\cdot)] + \delta F_i[\delta u(\cdot)] = F_i + \int_0^T w_i(t) \delta u(t) dt = 0 (\leq 0), \quad (8)$$

$$i = 1, 2, \dots, m,$$

$$\delta u(t) \in \delta U(t) \text{ при всех } t \in [0, T]. \quad (9)$$

Решение этой задачи позволяет осуществить основной шаг процесса — переход к управлению $u(\cdot) + \delta u(\cdot)$, причем (9) обеспечивает выполнение геометрического ограничения $u \in U$ в процессе поиска, (8) — выполнение дополнительных условий (3) с точностью до $O(\|\delta u\|^2)$ и отсутствие накопления этих погрешностей, связанных с нелинейностью задачи. Наконец, (7) обеспечивает максимальное понижение F_0 при переходе от u к $u + \delta u$.

К о н е ч н о м е р н а я а п п р о к с и м а ц и я. Отрезок $[0, T]$ разбивается на некоторое (обычно $\sim 10^2$) число малых счетных отрезков введением равномерной (например) сетки

$$0 = t_0 < t_1 < t_2 < \dots < t_n < \dots < t_N = T, \quad t_n = n\tau, \quad \tau = T/N. \quad (10)$$

В качестве искомой функции $u(\cdot)$ в расчетах фигурирует кусочно постоянная функция

$$u(t) = u_{n+1/2} \quad \text{при} \quad t_n < t < t_{n+1}, \quad n = 0, 1, \dots, N-1. \quad (11)$$

Вариация $\delta u(t)$ будет искаться в том же классе (11) кусочно постоянных функций.

Интегрирование системы (1) осуществляется подходящим стандартным методом; обычно шаг численного интегрирования меньше шага τ сетки (10). Единственная предосторожность, которую следует иметь в виду, связана с тем, что каждый узел t_n сетки (10) является возможной точкой разрыва $u(t)$, поэтому при численном интегрировании (1) в каждом отрезке $[t_n, t_{n+1}]$ должно содержаться целое число шагов интегрирования (1): если разрывы функции $u(t)$ оказываются внутри дискретных интервалов численного интегрирования (1), методы интегрирования высокого порядка точности эту точность теряют и превращаются в методы первого порядка. При интегрировании (1) траектория $x(t)$ запоминается в виде значений

$$x_n = x(t_n), \quad n = 0, 1, \dots, N.$$

Вычисление функциональных производных требует $(m+1)$ -кратного интегрирования систем вида

$$\frac{d\psi}{dt} + f_x^*[t]\psi = -Y[t], \quad \Gamma_x^*\psi = 0. \quad (12)$$

Эта система определена на невозмущенной траектории; обычно матрицы f_x , f_u запоминаются в виде последовательности f_u , $f_x\left[\frac{t_n + t_{n+1}}{2}\right]$.

Вообще, интегрирование (12), учитывая приближенный характер задач (7)–(9), обычно осуществляется менее точно, чем интегрирование основного уравнения (1). Заметим, что если, как это часто бывает, функции $f(x, u)$ имеют не очень простую ана-

литическую форму, расход времени на вычисление функциональных производных $w_i(t)$ в основном определяется вычислением матриц f_x, f_u и сравнительно слабо зависит от m — числа условий (4). В задачах с очень простыми правыми частями $f(x, u)$ это не так.

Производные $w_i(t)$ запоминаются в виде таблицы

$$w_i^{(n+1/2)} = \int_{t_n}^{t_{n+1}} w_i(t) dt, \quad n = 0, 1, \dots, N-1, \quad i = 0, 1, \dots, m. \quad (13)$$

Конструкция $\delta U(t)$ тоже имеет «сеточный» вид, т. е. задается последовательностью конструкций $\delta U_{n+1/2}$, и задача определения $\delta u(t)$ приобретает вид

$$\min_{\delta u} \sum_{n=0}^{N-1} (w_0^{(n+1/2)}, \delta u_{n+1/2}) \quad (7^*)$$

при условиях

$$F_i + \sum_{n=0}^{N-1} (w_i^{(n+1/2)}, \delta u_{n+1/2}) = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \quad (8^*)$$

$$\delta u_{n+1/2} \in \delta U_{n+1/2}, \quad n = 0, 1, \dots, N-1. \quad (9^*)$$

Конструкция $\delta U_{n+1/2}$. Опуская для простоты индекс $n+1/2$, опишем применявшийся в наших расчетах способ описания δU . Мы предполагаем, что множество допустимых по условию $u(t) + \delta u(t) \in \delta U(t)$ вариаций образует выпуклый конус K_t в r -мерном пространстве (r — размерность u). Как известно, выпуклые конусы допускают два способа описания: либо как пересечение некоторого набора полупространства, либо как выпуклая оболочка набора векторов. Именно этот второй способ и оказывается наиболее удобным.

Итак, множество δU опишем в виде

$$\delta u = \sum_{i=1}^k s_i e^i, \quad s_i^- \leq s_i \leq s_i^+, \quad (14)$$

где e^i — набор векторов в r -мерном пространстве, k — некоторое число, которое может быть как меньше, так и больше r в зависимости от геометрии U . Числа s_i^-, s_i^+ обеспечивают невыход $u + \delta u$ за пределы U и задают размеры области δU в соответствии с требованиями а), б), в), г) (стр. 166). Требование $u + \delta u \in U$ обычно учитывается легко и после решения несложных геометрических задач дает оценки для s_i^-, s_i^+ , учет остальных требований подробно обсуждается ниже. Мы не будем стараться сформулировать общие способы выбора набора e^i , предпочитая пояснить суть дела на нескольких характерных примерах.

Пусть U имеет вид, изображенный на рис. 16 (такого типа области U использовались в задаче выбора оптимального состава защиты от излучения; см. § 33). В зависимости от положения u в U , наборы e и s^+ , s^- следующие:

$$\begin{aligned} A: \quad e^1 &= \{1; -1; 0\}, \quad s_1^- = 0; \quad s_1^+ > 0, \\ e^2 &= \{0; -1; 0\}, \quad s_2^- = 0; \quad s_2^+ > 0, \\ e^3 &= \{0; -1; 1\}, \quad s_3^- = 0; \quad s_3^+ > 0. \end{aligned}$$

Если точка u занимает положение B , близкое к угловой точке области U , естественно взять набор

$$\begin{aligned} B: \quad e^1 &= \{1; -1; 0\}; \quad s_1^+ > 0; \quad s_1^- > (u_1 - 1), \\ e^2 &= \{-1; 0; 1\}; \quad s_2^- = 0; \quad s_2^+ > 0, \\ e^3 &= \{-1; 0; 0\}; \quad s_3^- = 0; \quad s_3^+ > 0. \end{aligned}$$

Если область имеет криволинейную границу, набор векторов может строиться так, как это показано на рис. 17; здесь число k векторов

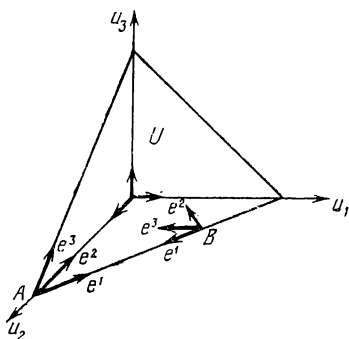


Рис. 16.

$$\begin{aligned} U: \quad u_1 &\geq 0, \quad u_2 \geq 0, \quad u_3 \geq 0, \\ u_1 + u_2 + u_3 &\leq 1. \end{aligned}$$

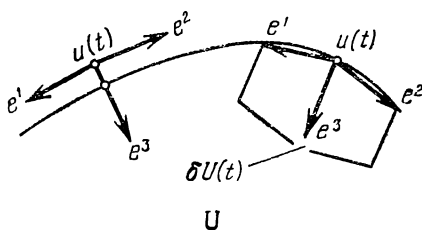


Рис. 17.

e превосходит размерность управления; это сделано для того, чтобы предупредить вырождение области δU и не сузить множества возможных направлений смещения u .

В принципе допустим выход u за пределы U (см. рис. 17); положив $s_3^- > 0$, мы включим в процесс элемент корректировки условия $u(t) \in U$.

Окончательная форма задачи отыскания $\delta u(t)$. Итак, на каждом интервале (t_n, t_{n+1}) конструкция $\delta U_{n+1/2}$ имеет вид

$$\delta u \in \delta U_{n+1/2}; \quad \delta u = \sum_{i=1}^k s_i e^i, \quad s_i^- \leq s_i \leq s_i^+$$

(разумеется, e^i , s_i^- , s_i^+ , s_i следует пометить еще индексом $n + 1/2$; ради простоты он опущен).

Используя эту конструкцию, имеем

$$(w^{n+1/2}, \delta u) = \sum_{i=1}^k s_i (w^{n+1/2}, e^i).$$

Теперь задача (7)–(9) превращается в задачу линейного программирования:

найти числа s_n , $n = 1, 2, \dots, N$, из условий

$$\begin{aligned} \min_{\{s_n\}} \sum_{n=1}^N s_n h_n^0, \\ X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \\ s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N. \end{aligned} \quad (15)$$

Во избежание недоразумений заметим, что в (15) индекс n имеет другой смысл, не совпадающий с его смыслом в (7)–(9); числа h_n^i очевидным образом связаны со скалярными произведениями (w_i, e^j) . Заметим, что в расчетах система векторов для счетных интервалов (t_n, t_{n+1}) не запоминается, после решения задачи (15) эта система вновь восстанавливается, когда из чисел s_n строится новое управление

$$u_{n+1/2} = u_{n+1/2} + \sum_{i=1}^k s_i e^i.$$

Задача (15) является стандартной задачей линейного программирования, имеющей, однако, специфическое происхождение: она возникла при сеточной аппроксимации непрерывной задачи типа линейного программирования (7)–(9). Поэтому, например, $N \gg m$ (в расчетах автора $N \sim 10^2 \div 10^3$, $m \sim 1 \div 10$). Решение ее стандартными методами, например, симплекс-методом, может привести к неоправданно большим затратам машинного времени: в этих алгоритмах фундаментальную роль играют m -мерные «границы» — множества точек $(m+1)$ -мерного пространства.

$$x = X + \sum_{n=1}^N s_n h_n, \quad X = \{0, X^1, \dots, X^m\}, \quad h_n = \{h_n^0, h_n^1, \dots, h_n^m\},$$

получающиеся при закреплении всех, за исключением некоторых m , чисел s_n в положении s_n^- или s_n^+ . Основной шаг симплекс-метода состоит в переходе от некоторой такой грани к смежной, в которой набор m «свободных» s_n изменяется на единицу (см. § 47). При больших N , когда шаг сетки $\tau \rightarrow 0$, эти грани вырождаются в точки, и метод, тщательно отслеживающий эти грани, становится

все менее и менее эффективным. Происхождение задачи (15) из непрерывной сказывается еще и в очень сильной почти-вырожденности задачи: векторы h_n на соседних интервалах отличаются друг от друга на $O(\tau^2)$ (заметим, что $\|h\| \sim O(\tau)$)*), причем по мере приближения управления к оптимальному вырожденность задачи часто еще более усиливается. Напомним, что задача линейного программирования называется *невыврожденной*, если для любого вектора $g = \{g^0, g^1, \dots, g^m\}$ равенство $(h_n, g) = 0$ имеет место не более, чем для m векторов h_n ; в противном случае задача вырождена. Известно, что в вырожденных задачах возможно так называемое «зацикливание» алгоритма симплекс-метода; разработаны методы борьбы с незначительной вырожденностью, когда число равенств $(h_n, g) = 0$ близко к m . В задаче же (15) мы сталкиваемся с ситуацией, когда для большого, сравнимого с N , числа индексов n справедлива оценка: $|(h_n, g)| \ll \|h_n\| \|g\|$; и если точная вырожденность может привести к отсутствию сходимости алгоритма, то мощная «почти-вырожденность» серьезно замедляет сходимость. Поэтому для решения задачи (15) следует использовать алгоритмы, так сказать, выдерживающие предельный переход от конечномерной задачи (15) к непрерывной (7)–(9). В § 48 мы описываем разработанный и применяемый автором метод.

З а м е ч а н и е о л и н е й н ы х з а д а ч а х о п т и м а л ь н о г о у п р а в л е н и я. Легко заметить, что определяющая вариацию $\delta u(t)$ задача (7)–(9) есть простая линейная задача оптимального управления для системы

$$\begin{aligned} x^i &= (w_i(t), \delta u(t)), \quad x^i(0) = F_i, \quad i = 0, 1, \dots, m, \\ 0 &\leq t \leq T; \quad \delta u(t) \in \delta U(t), \end{aligned} \quad (16)$$

в которой требуется минимизировать $x^0(T)$ при условиях

$$x^i(T) = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m.$$

Поэтому можно было бы, не разрабатывая специальных алгоритмов для (15), использовать методы решения линейных задач. По мнению автора, наиболее эффективным направлением в разработке методов решения линейных задач является их конечномерная сеточная аппроксимация, сведение к задаче линейного программирования и решение последней подходящим, учитывающим происхождение задачи алгоритмом. Например, если бы мы попытались решать задачу (16) методом поворота опорной гиперплоскости, то, по существу, это и был бы метод, описанный в § 48, но без весьма существенного элемента — процедуры $\min \|x\|_G$ (см. § 48), роль которой в эффективности процесса, без преувеличения, — решающая. Велика роль этой процедуры и в решении строго выпуклой задачи квадратического программирования

*) $\|h\|$ того же порядка, что и $\|w_i\|$, а $\|w_i\| = O(\tau)$ (см. (13)).

(алгоритм ее решения описан в § 49), хотя сходимость метода поворота опорной гиперплоскости доказывается строго и без нее.

З а м е ч а н и е о г е о м е т р и ч е с к о м о г р а н и ч е н и и. Входящее в задачу условие $u \in U$ обычно реализуется заданием одного или нескольких неравенств

$$\varphi_j [u(t)] \leq 0 \quad \text{при всех } t; \quad j = 1, 2, \dots, J. \quad (17)$$

Линеаризуя (17), получим ограничения для $\delta u(t)$:

$$\varphi_j [u(t)] + \frac{\partial \varphi_j}{\partial u} \delta u(t) \leq 0 \quad \text{при всех } t, \quad j = 1, 2, \dots, J. \quad (18)$$

Они могут быть включены в задачу линейного программирования непосредственно, без построения системы векторов e^t и без решения каких бы то ни было

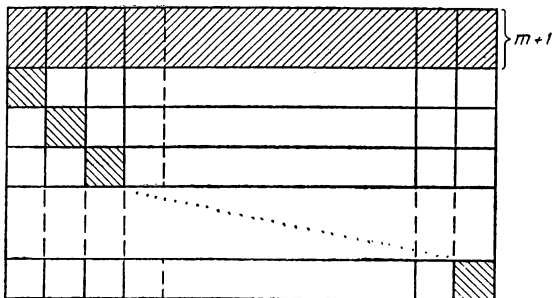


Рис. 18.

геометрических задач. Разумеется, следовало бы добавить к (18) простые ограничения на компоненты вариаций δu типа $|\delta u| \leq s$, чтобы не выходить за рамки точности линейного приближения. Действуя таким образом, мы получим для определения δu задачу линейного программирования типа (15), однако матрица этой задачи будет иметь блочную структуру, схематически изображенную на рис. 18, где заштрихованы ненулевые части матрицы (двусторонние ограничения $-s \leq \delta u \leq s$, естественно, в матрицу не включаются).

Первые $(m+1)$ строк матрицы совершенно аналогичны всей матрице задачи (15); условия (18) порождают большое, $\sim T/\tau$, число малых блоков, каждый из которых связывает искомые компоненты управления, относящиеся к одному интервалу счетной сетки (мы не обсуждаем здесь очевидных возможных упрощений задачи, связанных с тем, что условие $\varphi_j(u) \leq 0$ следует учитывать лишь при тех t , где $\varphi_j[u(t)] \simeq 0$). Итак, получена задача, вертикальный размер которой может быть очень велик, порядка N . Хорошо известно, что трудность задачи линейного программирования в основном определяется именно ее вертикальным размером. Рекомендовать для решения этой задачи стандартный симплекс-метод едва ли возможно. Не исключено, что его специальные итерационные варианты, рассчитанные на матрицы блочной структуры, окажутся приемлемыми, однако все это нуждается в экспериментальной проверке.

Именно по этой причине автор предпочел конструкцию вариации управления типа (14). Требуя для своего формирования решения некоторых геометрических задач, она приводит к задаче линейного программирования, вертикальный размер которой равен числу входящих в задачу функционалов $F_i [u(\cdot)]$.

§ 20. Метод последовательной линеаризации. Вычислительная технология

Общие соображения о построении минимизирующей последовательности управлений не определяют однозначно алгоритм фактического решения прикладных задач. Если предполагается такой алгоритм все-таки довести до программы, возникает большое число вопросов, которые мы условно отнесем к вычислительной технологии. Грамотное и ответственное решение таких вопросов совершенно необходимо; неудачное или случайное их решение способно испортить даже очень хорошую общую идею. Однако специфической особенностью этой стороны дела является отсутствие однозначного решения вопросов технологии.

В настоящем параграфе будут рассмотрены именно такие вопросы. Не претендуя на окончательное однозначное и наилучшее решение их, автор ставил следующие цели:

1) достаточно четко выделить основные элементы вычислительной технологии;

2) сформулировать общие качественные соображения, которые используются при их решении (обычно с этим тесно связан анализ тех вычислительных затруднений, которые возникают при заведомо «плохом» решении вопросов технологии).

Конкретнее, речь идет о следующем: в вычислительный алгоритм входит некоторое число параметров, значения которых очень существенны для эффективности метода. К сожалению, «правильные» числовые значения этих параметров различны для разных задач и даже для разных этапов решения одной и той же задачи, так что проявить при их выборе здравый смысл или использовать опыт решения предыдущей задачи не так-то просто. Введение в эти вопросы некоторого элемента объективности в практической работе автора осуществлялось следующим образом: подбор непосредственно входящих в алгоритм параметров осуществлялся некоторыми механизмами обратной связи, основанными на анализе фактического хода процесса поиска оптимального управления. Эти механизмы также неоднозначны и включают в себя некоторые параметры. Однако эти последние уже слабо зависят от конкретной задачи, имеют обычно простой и наглядный содержательный смысл. При назначении этих параметров сравнительно нетрудно проявить здравый смысл и использовать предшествующий опыт. Ниже эта сторона вопроса будет подчеркиваться.

Н о р м и р о в к а з а д а ч и. Входящие в постановку задачи функционалы $F_i [u(\cdot)]$ обычно имеют разный физический смысл и разные размерности; задача допускает очевидное эквивалентное преобразование $F_i \rightarrow \kappa_i F_i$, которое, не меняя существенно существа дела, имеет самые серьезные последствия с точки зрения фактического хода процесса поиска. Вопрос о разумной нормировке задачи (о выборе чисел κ_i) тесно связан со следующим: какие приращения функционалов ΔF_i следует считать равноценными? Ответ может быть примерно таким: те, которые порождены одинаковыми вариациями управления; именно это соображение и будет использовано. Другое соображение состоит в том, что эти равноценные приращения функционалов должны выражаться числами одного порядка. Реализуется же подходящая нормировка на стадии решения задачи линейного программирования

$$\min_{\{s_n\}} \sum_{n=1}^N s_n h_n^0, \\ X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \quad (1) \\ s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N.$$

Сначала величины h_n^i вычисляются в тех единицах, в которых заданы функционалы F_i при содержательной постановке задачи. Перед решением задачи (1) она нормируется — каждая строка ($i = 0, 1, \dots, m$) делится на $\left\{ \sum_{n=1}^N (h_n^i)^2 \right\}^{1/2}$. В этом случае задача линейного программирования инвариантна относительно выбранных первоначально единиц измерения функционалов F_i . Такая нормировка задачи (1) полезна с точки зрения использовавшегося метода ее решения (см. § 48); одним из существенных элементов его является минимизация квадратичной формы

$$\min_{\{s\}} \sum_{i=0}^m \left\{ X^i + \sum_{n=1}^N s_n h_n^i \right\}^2 = \min_s \|x(s)\|^2,$$

и в этом случае очень важно, чтобы компоненты вектора x были величинами одного порядка. В противном случае, если одна из компонент существенно больше остальных, в процессе минимизации $\|x(s)\|^2$ содержательно несущественные, но выражаемые большим числом невязки в этой компоненте «забивают» остальные, содержательно важные, но выражаемые очень маленькими числами. Разумеется, по существу эффективность различных методов минимизации $\|x(s)\|^2$ определяется более тонкими факторами, например, разбросом собственных чисел некоторых матриц, образованных наборами векторов h_n . Однако учет этих тонких обстоятельств был бы слишком сложным.

Изложенные выше соображения совершенно аналогичны приемам, используемым в методе Ньютона (см. § 43). Подчеркнем, что вопрос о выборе рациональных единиц измерения различных функционалов, решение которого неясно а priori, решается на основании анализа объективных характеристик — числовых значений функциональных производных $\partial F_i[u(\cdot)]/\partial u(\cdot)$. Именно это обстоятельство представляется нам существенным, сам же способ нормировки хотя и естествен, но достаточно условен и связан с характером дальнейшей работы с этими производными, т. е., по существу, с используемым методом решения задачи (1).

Читатель, видимо, уже заметил, что числовые величины функциональных производных зависят не только от единиц измерения F_i , но, если управление есть вектор-функция, и от единиц измерения различных ее компонент.

Последние тоже часто имеют разные физические размерности, и первоначальная постановка задачи производится в некоторой произвольной системе единиц. Возникает вопрос о выборе их — причем решен он должен быть в первую очередь, до нормировки задачи (1).

Н а ч а л ь н ы й в ы б о р х а р а к т е р н ы х в е л и ч и н в а р и а ц и й к о м п о н е н т у п р а в л е н и я. Рассмотрим соображения, используемые при выборе единиц измерения компонент управления и характерных величин их вариаций. Этими величинами определяются размеры окрестности δU , в которой решается линеаризованная задача (см. § 19). Основное качественное соображение, которое здесь используется, состоит в выявлении максимальных вариаций компонент управления, при которых линеаризация задачи имеет некоторую предписанную точность. Разумеется, эти величины определяются с невысокой точностью, но в данном случае этого достаточно. Конкретно эти соображения реализуются так. Перед решением задачи производится прикидочный расчет: исходное управление $u(\cdot)$ подвергается пробным вариациям отдельно по каждой компоненте, т. е. проводится серия расчетов траектории с управлением $u_r(t)$, $u_r(t) + \Delta$, $u_r(t) + 2\Delta$, ... (остальные компоненты u в данной серии прикидок остаются фиксированными). Вычисляя фактические приращения функционалов и сравнивая их с предсказанными на основе линейной теории возмущений, нетрудно установить характерную величину $\Delta^{(r)}$ возмущения r -й компоненты управления, при которой формулы первого порядка точности

$$\delta F_i[\delta u(\cdot)] = \int_0^T w_i(t) \delta u(t) dt$$

дают совпадение с фактическими приращениями ΔF_i (для всех i) с точностью, допустим, 10—20%. Прделав такие расчеты

для каждой компоненты управления и найдя числа $\Delta^{(r)}$, мы устанавливаем как естественные единицы измерений для разных компонент u (с тем, чтобы в этих новых единицах $\Delta^{(r)}$ были числами примерно равной величины), так и характерные масштабы вариаций управления.

З а м е ч а н и е. Не следует бояться кажущейся громоздкости этой «предварительной разведки»: ведь обычно в прикладных задачах управление имеет не слишком высокую размерность (в расчетах автора — не больше трех-четырёх; во всех известных автору чужих расчетах размерность u не превосходит двух). Кроме того, начинается эта прикидка отнюдь не с произвольных величин — некоторые предварительные суждения о величинах $\Delta^{(r)}$, как правило, имеются у вычислителя, занимающегося решением конкретной задачи. Однако одно предостережение следует сделать. Всякий вычислитель, имевший дело с задачами поиска минимума, против сделанных выше рекомендаций выдвинет возражения примерно такого характера:

а) информация о соотношении величин $\Delta^{(r)}$ получена на исходном управлении $u(\cdot)$; в процессе поиска $u(\cdot)$ изменится, и эта информация потеряет ценность;

б) при установлении $\Delta^{(r)}$ были использованы очень специфические вариации управления, «разведка» произведена, так сказать, лишь по одному случайному направлению в пространстве управлений; между тем, как показывает опыт, дифференциальные свойства функционалов $F_i[u(\cdot)]$ чрезвычайно сильно зависят от направления варьирования управления, и именно с этим связана трудность задачи поиска минимума.

Эти возражения и правильны, и неправильны. Точнее, они правильны, если мы имеем дело с негладкими задачами, но для задач гладких они неверны. Опыт решения автором большого числа разнообразных прикладных задач и анализ задач, решенных другими, показали, что, как правило, естественная постановка прикладной задачи оптимального управления делается в терминах очень гладких функционалов, для которых формулы линейного приближения

$$\delta F_i[\delta u(\cdot)] = \int_0^T w_i(t) \delta u(t) dt$$

имеют хорошую точность при достаточно больших $\|\delta u(\cdot)\|$, и эта точность более или менее одинакова во всех направлениях в функциональном пространстве управлений. Противоречащие этому примеры обычно связаны с «неестественными» постановками задач, в которых ради упрощения аналитической формы задачи используются приемы типа «штрафных функций». Именно таким приемам мы обязаны появлению в задачах оптимального управления чрезвычайно капризных функционалов, поведение

которых очень различно при смещении управления по различным направлениям. В § 18 подробно рассматриваются дефекты некоторых часто рекомендуемых приемов упрощения задачи.

К сожалению, это внешнее упрощение сопровождается ухудшением дифференциальных свойств задачи. Автор в своих расчетах никогда не использовал подобных «упрощений» и следил за тем, чтобы они не использовались и «физиками», предлагающими ему ту или иную содержательную задачу. Разумеется, все эти утверждения не следует абсолютизировать, они отражают прежде всего точку зрения и опыт автора.

Итак, и в вопрос о неясном заранее естественном выборе единиц измерения для разных компонент u внесен элемент объективности. Что же произойдет, если естественные пропорции между величинами вариаций разных компонент сильно нарушены? Рассмотрим, несколько утрируя, задачу с двумя управляющими функциями $u_1(t)$, $u_2(t)$, и пусть процесс поиска ведется с шагом по u_2 столь малым, что связанная с $\delta u_2(t)$ вариация функционалов задачи

$$\delta F = \int_0^T w^{(2)}(t) \delta u_2(t) dt$$

имеет численное значение порядка квадратичной ошибки, возникающей при использовании вариации δu_1 :

$$F[u(\cdot) + \delta u(\cdot)] = F[u(\cdot)] + \int_0^T w(t) \delta u(t) dt + O(\|\delta u_1\|^2),$$

и

$$\int_0^T w^{(2)}(t) \delta u_2(t) dt \simeq O(\|\delta u_1\|^2).$$

В этом случае сравнительно быстро будет найдено управление, оптимальное лишь по одной компоненте $u_1(t)$ управления, компонента же $u_2(t)$ при этом не успеет сколько-нибудь заметно отклониться от исходного состояния. При дальнейших вариациях управления эволюция компоненты $u_2(t)$ будет определяться не столько стремлением понизить значение $F_0[u(\cdot)]$, сколько необходимостью погашать невязки в дополнительных условиях $F_i[u(\cdot)] = 0$, $i=1, \dots, m$, возникающие при вариации δu_1 и имеющие порядок $O(\|\delta u_1\|^2)$.

Регулирование шага в процессе поиска. Описанная выше процедура прикидочных расчетов позволяет выбрать ориентировочные значения для величин вариаций разных компонент управления. Следует предусмотреть некоторый алгоритм корректировки величины вариации в процессе поиска.

Этот алгоритм имеет характер механизма обратной связи и основан на сравнении величин вариаций функционалов $\delta F_i[\delta u(\cdot)]$, рассчитанных по формулам линейной теории возмущений

$$\delta F_i[\delta u(\cdot)] = \int_0^T w_i(t) \delta u(t) dt$$

с их фактическими приращениями

$$\Delta F_i \equiv F_i[u(\cdot) + \delta u(\cdot)] - F_i[u(\cdot)].$$

Последние находятся после вариации управления $u(\cdot) \rightarrow u(\cdot) + \delta u(\cdot)$, интегрирования системы дифференциальных уравнений $\dot{x} = f(x, u + \delta u)$ и вычисления на новой траектории значений функционалов $F_i[u(\cdot) + \delta u(\cdot)]$. Сравнение δF_i с ΔF_i позволяет судить о том, не является ли используемый шаг по управлению слишком малым: если совпадение ΔF_i с δF_i излишне точно, шаг следует увеличить. Если совпадение ΔF_i с δF_i слишком грубо — шаг уменьшается. Не претендуя на наилучшее решение вопроса, но лишь для того, чтобы быть конкретнее, укажем на используемое автором критерий того, что есть хорошее и что есть плохое совпадение ΔF с δF . Обычно расхождение δF и ΔF менее, чем на 10% считалось очень малым и приводило к увеличению шага. Расхождение более чем на 30% считалось большим и влекло за собой уменьшение шага. Границы 10% и 30%, разумеется, достаточно условны.

Мы не будем здесь давать окончательных конкретных конструкций алгоритмов вычисления величин s_n^- , s_n^+ , входящих в задачу линейного программирования и определяющих размеры и форму окрестности $\delta U(t)$. Эти алгоритмы носят достаточно произвольный характер, и изложение их в расчете на самый общий случай едва ли целесообразно. Конкретные формы этих алгоритмов будут описаны для некоторых частных задач (см., например, § 34), здесь же мы ограничимся изложением некоторых достаточно общих соображений, используемых в этих алгоритмах.

1. Обычно величина вариации управления определялась одним числом S . Используя это число, тот или иной конкретный алгоритм (см., например §§ 29, 34) рассчитывает числа s_n^- , s_n^+ . Можно, для определенности, иметь в виду самый простой способ: пусть описанный выше прикидочный расчет определил соотношение величин вариаций по разным компонентам управления u : x_1, x_2, \dots, x_r (r -размерность u). Тогда можно считать максимальной допустимой вариацией j -й компоненты u на n -м интервале временной сетки величину Sx_j . В расчетах обычно использовались более сложные алгоритмы, учитывающие некоторые соображения о ценности вариации управления на том или ином интервале временной сетки, с тем чтобы допустить большую вариацию

там, где она более эффективна с точки зрения решаемой задачи; конкретные реализации подобных соображений см. в §§ 29, 34, содержащих описание опыта решения отдельных прикладных задач. Регулирование шага осуществлялось увеличением или уменьшением этого числа S . Обычно это осуществлялось пересчетом S по формуле $S := 0,8S$, если δF_i плохо совпадают с ΔF_i , и $S := 1,15S$, если совпадение δF_i с ΔF_i хорошее. Заметим, что мы предпочитаем сравнительно осторожную тактику подбора числа S в процессе поиска. Подобные алгоритмы подбора шага использовались многими, однако, как правило, с более решительными реакциями $S := S/2$ или $S := 2S$. В § 29 подробно показан процесс подбора S при решении задачи о вертикальном подъеме ракеты. В экспериментальных целях начальное значение S бралось в 10 раз большим и в 10 раз меньшим той естественной величины, которая определялась прикидочным расчетом. Приведенная в § 29 таблица показывает, как довольно быстро вырабатывается практически одна и та же величина S .

2. Следует сделать определенные уточнения относительно того, как нужно сравнивать величины δF_i и ΔF_i , поскольку естественное определение относительной ошибки формулой типа

$$2 \frac{|\delta F_i - \Delta F_i|}{|\delta F_i + \Delta F_i|} \quad (2)$$

может привести к ошибочным заключениям. Дело в следующем: обычно в расчетах величины $F_i[u(\cdot)]$, $i=1, 2, \dots, m$, суть величины порядка $O(\|\delta u\|^2)$, они появляются вследствие нелинейности задачи. Выбирая вариацию управления $\delta u(t)$, мы выполняем, в частности, условия вида

$$F_i[u(\cdot)] + \delta F_i[\delta u(\cdot)] = F_i + \int_0^T w_i(t) \delta u(t) dt = 0. \quad (3)$$

Таким образом, $|\delta F_i| \simeq |F_i| \simeq O(\|\delta u\|^2)$. Такой же величиной будет и ΔF_i . Поэтому вместо формулы (2) использовался следующий критерий точности линейного приближения: запишем выражение

$$F_i[u(\cdot) + \delta u(\cdot)] = F_i[u(\cdot)] + \int_0^T w_i \delta u dt + \xi_i \int_0^T |w_i \delta u| dt, \quad (4)$$

где ξ_i — параметр, подбираемый так, что равенство (4) выполняется; это число ξ_i и считается относительной точностью линейного приближения для функционала F_i при используемой величине вариации управления. В формуле (3) по смыслу решаемой задачи происходит взаимная компенсация положительных и отрицательных слагаемых, поэтому формула (2) непригодна для оценки точности линейного приближения.

3. Приведем теперь грубые качественные соображения, по которым точность линейного приближения $\sim 30\%$ считается удовлетворительной. Пусть вариация управления, определяемая величиной S , обеспечивает 30% точности, и предсказанное понижение

$$\delta F_i = \int_0^T w_0(t) \delta u(t) dt$$

отличается от фактического на 30% , т. е., например,

$$\Delta F_0 = 0,7 \delta F_0.$$

Пусть уменьшение S вдвое практически восстанавливает точность линейного приближения, т. е.

$$\delta \bar{F}_0 = \int_0^T w_0(t) \delta \bar{u}(t) dt = \Delta F_0,$$

где $\delta \bar{u}(t)$ — вариация управления, соответствующая шагу $\bar{S} = \frac{1}{2} S$. Тогда $\delta \bar{u}(t) \simeq 0,5 \delta u(t)$, $\delta \bar{F}_0 \simeq 0,5 \delta F_0$ и $\Delta \bar{F}_0 = 0,5 \delta F_0$. Таким образом, один шаг процесса минимизации с вариацией величиной S дал выигрыш в F_0 величиной $0,7 \delta F_0$, а при тех же затратах машинного времени шаг процесса в условиях большей точности линейного приближения, обеспечиваемой величиной допустимой вариации $\bar{S} = \frac{1}{2} S$, дал выигрыш в $F_0 \sim 0,5 \delta F_0$; для нас же основным критерием является величина понижения F_0 за один шаг процесса. Разумеется, эти соображения имеют ценность в основном в начальной стадии процесса поиска минимума, когда происходит выход $u(\cdot)$ в окрестность искомого оптимального управления. На заключительной стадии процесса поиска, когда происходит уточнение управления, сопровождающееся незначительным понижением F_0 , требования к точности линейного приближения обычно повышаются.

§ 21. Метод последовательной линеаризации.

Задачи с функционалами, дифференцируемыми по Гато

Здесь будут описаны разработанные автором и использованные в прикладных расчетах приемы, позволяющие с приемлемыми затратами машинного времени эффективно решать задачи, в формулировку которых входят функционалы следующих типов:

$$F[u(\cdot)] \equiv \max_t \Phi[x(t)], \quad (1)$$

$$F[u(\cdot)] \equiv \max_t \Phi[x(t), u(t)], \quad (2)$$

$$F[u(\cdot)] \equiv \int_0^T |\Phi[x(t), u(t)]| dt. \quad (3)$$

Эти функционалы в общем случае (а именно этот общий случай и реализовался во всех расчетах автора) не имеют производных Фреше; они дифференцируемы лишь по направлениям в функциональном пространстве (см. § 4). Это обстоятельство делает решение задач с подобными функционалами очень сложным. Конструкции (1)–(3) охватывают большинство возникающих в приложениях функционалов, дифференцируемых лишь по Гато. Каждая конструкция требует специфического подхода и мы опишем их по отдельности.

Задачи с функционалами типа $\max_t \Phi[x(t)]$. Пусть функционал типа (1) входит в дополнительное условие, т. е. на искомое оптимальное управление наложено ограничение, имеющее вид ограничения в фазовом пространстве

$$F[u(\cdot)] \equiv \max_t \Phi[x(t)] \leq 0. \quad (4)$$

Разумный выбор вариации управления теперь стеснен условием вида

$$F[u(\cdot) + \delta u(\cdot)] \leq 0, \quad (5)$$

которое, к сожалению, не имеет простой формы: главная (по порядку $\|\delta u\|$) часть приращения функционала (1) не есть линейный функционал. При решении подобных задач успешно использовался прием аппроксимации (5), описанный в общих чертах в § 8. Именно, после интегрирования системы $\dot{x} = f(x, u)$ с невозмущенным управлением в узлах сетки $\{t_n\}_{n=1}^N$ вычислялась функция $\Phi_n = \Phi[x(t_n)]$ и выделялось подмножество узлов сетки M условием

$$t_n \in M, \text{ если } \Phi_n > F[u(\cdot)] - \varepsilon |F|, \quad (6)$$

$$\varepsilon \sim 0,05, \quad F[u(\cdot)] = \max_n \Phi_n.$$

Далее, из этого множества узлов M выбиралось k точек аппроксимации t^1, t^2, \dots, t^k и полагалось

$$F[u(\cdot) + \delta u(\cdot)] \simeq \max_{j=1, \dots, k} \{\Phi[t^j] + \Phi_x \delta x(t^j)\}. \quad (7)$$

Что касается $\Phi_x \delta x(t^j)$, то это — линейный функционал от $\delta u(\cdot)$, и k раз решив задачу

$$\dot{\psi} + f_x^*[t]\psi = -\delta(t - t^j)\Phi_x[x(t^j)], \quad \Gamma_x^*\psi = 0, \quad j = 1, 2, \dots, k, \quad (8)$$

условие на вариацию (5) аппроксимируем k неравенствами

$$\Phi[x(t^j)] + \int_0^T w^{(j)}(t) \delta u(t) dt \leq 0, \quad j = 1, 2, \dots, k. \quad (5^*)$$

Эти условия уже очевидным образом включаются в задачу линейного программирования (19.15). Нужно только иметь в виду, что

дифференцируемый по Фреше функционал порождает лишь одну строку в матрице задачи линейного программирования и требует лишь однократного решения задачи типа (8). Что касается точек аппроксимации, то они выбираются в сравнительно небольшом числе (в расчетах автора $k=3, 4$; большее число бралось лишь в методических задачах), однако, разумеется, не фиксируются, а размещаются в зависимости от профиля функции $\Phi[x(t)]$ на данном этапе вычислительного процесса. Именно эта «подвижность» точек аппроксимации позволяет обеспечить удовлетворительную точность условия $F[u(\cdot)] \leq 0$ при небольших k , хотя множество M при этом включало большое, $\sim N$, число узлов сетки. Если каждую точку $t_n \in M$ считать точкой аппроксимации, условие $F[u(\cdot)] \leq 0$ будет выполнено с точностью до $O(\tau)$ (см. § 8). Но то, что перемещающиеся от шага к шагу точки аппроксимации при небольшом числе их способны обеспечить аналогичный результат, уже не столь очевидно, и соответствующая теорема будет сейчас доказана.

Теорема 1. Пусть на интервале $[t', t''] \in [0, T]$ фазовая траектория $x(t)$ управляемой системы $\dot{x} = f(x, u)$, $\Gamma(x) = 0$ подчинена условию $\Phi[x(t)] \leq 0$ при всех $t \in [t', t'']$.

Пусть последовательные управления $u^{(v)}(\cdot)$ строятся процессом $u^{(v+1)}(\cdot) = u^{(v)}(\cdot) + \delta u^{(v+1/2)}(\cdot)$.

Пусть вариации $\delta u^{(v+1/2)}$ строятся следующим образом: интервал $[t', t'']$ разбивается на k равных частей, на каждой из них выбирается своя точка аппроксимации $t^{j,v}$, $j=1, 2, \dots, k$, как точка максимума $\Phi[x^{(v)}(t)]$ на j -й части $[t', t'']$. Вариацию $\delta u^{v+1/2}$ будем считать удовлетворяющей условиям

$$\Phi[x^v(t^{j,v})] + \Phi_x \delta x(t^{j,v}) \leq 0, \quad j=1, 2, \dots, k \quad (9)$$

(здесь $\delta x(t)$ — вариация фазы, порождаемая вариацией управления $\delta u^{v+1/2}$) и условию достаточной малости

$$|\delta u(t)| \leq s \quad \text{при всех } t \in [0, T], \quad (10)$$

где s — некоторое малое число, а (10) следует понимать как покомпонентное ограничение. В остальном $\delta u^{v+1/2}(t)$ — совершенно произвольна. Тогда при всех v и $t \in [t', t'']$

$$\Phi[x^v(t)] \leq \frac{Cs(t'' - t')}{k} + O(s^2). \quad (11)$$

Здесь C — постоянная, зависящая от величин производных $f(x, u)$, $\Phi_x[x(t)]$ на реализующихся траекториях; эти величины мы будем считать равномерно ограниченными, что соответствует практике решения прикладных задач.

Доказательство. Пусть $\delta u^{v+1/2}(\cdot)$ — вариация, удовлетворяющая (9) и (10), $u^{v+1}(\cdot) = u^v(\cdot) + \delta u^{v+1/2}(\cdot)$, и $x^{v+1}(t)$ — соответствующая фазовая траектория. Пусть

$$\Phi[x^{v+1}(t^*)] = \max_{t' \leq t \leq t''} \Phi[x^{v+1}(t)]. \quad (12)$$

Теорема будет доказана, если (11) верно в точке t^* . На расстоянии, не большем $(t'' - t')/k$, находится ближайшая к t^* точка аппроксимации из совокупности $\{t^j, v\}_{j=1}^k$; обозначим ее t_* . В соответствии с (9)

$$\Phi[x^{v+1}(t_*)] = \Phi[x^v(t_*)] + \Phi_x \delta x^{v+1/2}(t_*) + O(s^2) \leq O(s^2). \quad (13)$$

Предположим, для определенности, $t_* < t^*$ (обратный случай анализируется точно так же) и используем очевидную связь

$$\Phi[x^{v+1}(t^*)] = \Phi[x^{v+1}(t_*)] + \int_{t_*}^{t^*} R[x^{v+1}(t), u^{v+1}(t)] dt,$$

где

$$R(x, u) = \frac{d\Phi}{dt} = \Phi_x[x(t)] f[x(t), u(t)].$$

Для $\Phi[x^{v+1}(t_*)]$ имеем (13); оценим второе слагаемое

$$\begin{aligned} \int_{t_*}^{t^*} R(x^{v+1}, u^{v+1}) dt &= \int_{t_*}^{t^*} R(x^v + \delta x^{v+1/2}, u^v + \delta u^{v+1/2}) dt = \\ &= \int_{t_*}^{t^*} R(x^v, u^v) dt + \int_{t_*}^{t^*} (R_x \delta x + R_u \delta u) dt, \end{aligned}$$

но $\int_{t_*}^{t^*} R(x^v, u^v) dt = \Phi[x^v(t^*)] - \Phi[x^v(t_*)] \leq 0$, так как t_* и t^* принадлежат одной и той же части интервала $[t', t'']$, а t_* — точка максимума $\Phi[x^v(t)]$ на этой части.

Используя очевидную оценку

$$\left| \int_{t_*}^{t^*} (R_x \delta x + R_u \delta u) dt \right| \leq Cs(t^* - t_*),$$

получаем требуемый результат (14).

Из этой теоремы следует, что мы располагаем двумя ресурсами, обеспечивающими необходимую точность выполнения фазового ограничения: это число точек аппроксимации k и ограничение вариации управления s . Из (14) как будто следует, что можно вместо $\{k \text{ и } s\}$ взять $k' = 1$ и $s' = s/k$, и это обеспечит ту же точность. Это не так. Дело в том, что в условии теоремы неявно использовано еще одно важное предположение: считается, что на каждой

итерации задача линейного программирования, определяющая $\delta u(t)$, имеет решение. Но в эту задачу входят и величины невязок в условиях задачи. Если они достаточно велики, а числа s_n^- , s_n^+ — малы, задача линейного программирования не имеет решения и заменяется другой задачей — «терминальной» задачей погашения невязок, которая выполняется не за одну итерацию. Используя небольшое число точек аппроксимации, мы можем столкнуться с такой ситуацией, когда восстановление условия $\Phi[x(t)] \leq 0$ в точках аппроксимации сопровождается таким ростом величины $\Phi[x(t)]$ в каких-то других точках t , что в целом добиться необходимой точности выполнения условия $\Phi(x) \leq 0$ не удастся, и уменьшение шага s дела не меняет. Использовать полученные в теореме оценки для расчета необходимых величин k и s едва ли целесообразно; обычно подобные оценки грубы и дадут завышенное значение k (или заниженное значение s). К тому же теория не полна: нет оценок k и s , гарантирующих разрешимость задачи линейного программирования на каждой итерации.

Тем не менее, доказанная теорема полезна и используется в расчетах следующим образом: обычно в начале расчета полагается $k=2, 3$. Если процесс решения задачи соответствует предположению теоремы, т. е. на каждой итерации задача линейного программирования имеет решение, число k не меняется. Когда начинают встречаться случаи отсутствия решения задачи линейного программирования, k увеличивается на 1, пока не достигнет заданного предельного значения k^* . После этого начинается уменьшение s . Заметим еще, что этот механизм не работает в самом начале решения задачи, когда в силу грубости исходного управления $u^{(0)}(\cdot)$ все условия задачи грубо нарушены. Подробнее эти вопросы будут освещены при анализе решения конкретных задач (особенно в § 37).

Задачи на $\min_{u(\cdot)} \max_t \Phi[x(t)]$. Ограничимся для простоты задачами, в которой только минимизируемый функционал $F_0[u(\cdot)]$ имеет вид (1), а дополнительные условия сформулированы в терминах дифференцируемых по Фреше функционалов. Техника решения таких задач идентична только что описанной. Единственное отличие, заслуживающее пояснения, связано с формой задачи линейного программирования, определяющей вариацию управления. В задаче требуется найти s_n , $n=1, 2, \dots, N$, из условий

$$\begin{aligned} \min_s \max_{j=1, \dots, k} \left\{ X^{0,j} + \sum_{n=1}^N s_n h_n^{0,j} \right\}, \\ X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \\ s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N. \end{aligned} \quad (14)$$

В §§ 47, 48 алгоритмы решения задач линейного программирования изложены так, что решение задачи (14) осуществляется по общей схеме.

Задачи с функционалами $\max_t \Phi[x(t), u(t)]$.

Принципиальное отличие этого функционала от функционала (1) уже обсуждалось в § 4. Возможность с достаточной точностью аппроксимировать вариацию функционала (1) выражением (7) с небольшим числом k связана с гладкостью функции $\delta x(t)$, являющейся решением дифференциального уравнения в вариациях; следствием этого является и гладкость функции $\Phi_x[x(t)]\delta x(t)$, значения которой в окрестностях точек аппроксимации, грубо говоря, меняются при вариации управления в ту же сторону, в какую они меняются в точках аппроксимации. Поэтому, учитывая $\delta\Phi$ при построении $\delta u(\cdot)$ только в точках аппроксимации, мы в известной мере учитываем $\delta\Phi$ всюду, где $\Phi[x(t)] \sim \max \Phi[x(t)]$. Для функционала (2) это уже не так, $\delta u(t)$ — измеримая функция, ее значения в близких точках t' , t'' никак не связаны между собой, и аппроксимация типа (7) — неэффективна. Разумеется, она будет эффективна, если разместить по точке аппроксимации на каждом интервале счетной сетки (t_n, t_{n+1}) , входящем в множество M . Однако в проводившихся автором расчетах число таких интервалов было $\sim 10^2$, что уже приводит к задаче линейного программирования слишком тяжеловесной для того, чтобы решать ее на каждом шаге процесса построения минимизирующей последовательности управлений. Поэтому в расчетах использовался прием превращения компонент управления, явно входящих в функцию $\Phi[x, u]$, в фазовые координаты. Именно, полагаем

$$du/dt = v, \quad u(0) = \alpha. \quad (15)$$

Считая теперь искомым управлением функцию $v(\cdot)$ и параметр α , мы приходим к задаче, в которой функционал $F[v(\cdot), \alpha] \equiv \max_t \Phi[x(t), u(t)]$ уже является функционалом типа (1) и к нему

применима техника приближенного решения, изложенная выше. Однако, преодолев одни затруднения, замена (15) порождает другие.

1. Простые геометрические ограничения $u(t) \in U$ превращаются в ограничения в фазовом пространстве, т. е. в ограничения в терминах функционала типа (1); они учитываются так, как это было описано выше, что увеличивает размерность задачи линейного программирования (15, § 19). Правда, соответствующие элементы h_n^i матрицы этой задачи вычисляются просто: $h_n^i = (t_n - t_{n-1})$ левее соответствующей точки аппроксимации, $h_n^i = 0$ — правее ее. В целом замена (15) оказывается оправданной и позволяет эффективно решать прикладные задачи с хорошей точностью.

2. Более серьезные трудности связаны с тем, что замена (15) и соответствующая ей замена в вариациях

$$\frac{d\delta u}{dt} = \delta v(t), \quad \delta u(0) = \delta \alpha \quad (16)$$

означает сужение класса используемых вариаций: сеточный вариант решения уравнения (16)

$$\delta u(t_{n+1}) = \delta u(t_n) + \tau \delta v_{n+1/2}$$

является моделью малой функции класса Липшица

$$|\delta u(t_{n+1}) - \delta u(t_n)| \leq |t_{n+1} - t_n| C$$

с малой константой $C = \max_n |\delta v_{n+1/2}|$. Это приводит к определенным вычислительным трудностям, если искомое оптимальное управление $u(t)$ является, например, разрывной функцией. Получить разрыв в $u(t)$ — значит процессом малых вариаций $v + \delta v$ получить численный аналог δ -функции в $v(t)$. В принципе это возможно, однако, разумеется, процесс построения минимизирующей последовательности в терминах $v(\cdot)$ оказывается в этом случае довольно длительным. Подобные затруднения встретились при решении одной задачи с функционалом F_0 типа (2); они привели к тому, что процесс «застрял» достаточно далеко от оптимального управления и дал значение $F_0 = 1,700$ вместо точного минимума $F_0 = 1,585$.

В связи с этим был разработан и оказался очень эффективным следующий прием: вариация управления $\delta u(t)$ отсыкивалось в форме

$$\delta u(t) = \delta u'(t) + \delta u''(t).$$

При этом $\delta u'(t)$ — сеточная модель гладкой функции, определяемая произвольной малой сеточной функцией $\delta v(t)$ и уравнением (16), а $\delta u''(t)$ — сеточная модель измеримой функции, определяемая формально никак не связанными между собой значениями $\delta u_{n+1/2}$ на счетных интервалах (t_n, t_{n+1}) ; однако $\delta u''$ отлична от нуля лишь на тех интервалах (t_n, t_{n+1}) , которые не входят в выделенное множество M . В упомянутой задаче этот прием позволил получить четкий разрыв в $u(t)$ и значение $F_0 = 1,592$. Подробнее об этом см. в § 38.

Задачи с функционалом $\int_0^T |\Phi[x(t)]| dt$. Ограничимся для простоты задачей, в которой минимизируемый функционал F_0 имеет вид (3), остальные функционалы задачи дифференцируемы

по Фреше. Следуя анализу этой задачи в § 4, введем на $[0, T]$ множества M^- , M^0 , M^+ условиями

$$\begin{aligned} t \in M^-, & \text{ если } \Phi[x(t)] < -\varepsilon, \\ t \in M^0, & \text{ если } |\Phi[x(t)]| \leq \varepsilon, \\ t \in M^+, & \text{ если } \Phi[x(t)] > \varepsilon, \quad \varepsilon > 0, \end{aligned}$$

и запишем вариацию

$$\begin{aligned} \delta F_0[\delta u(\cdot)] = & \int_{M^+} \Phi_x[t] \delta x(t) dt - \int_{M^-} \Phi_x[t] \delta x(t) dt + \\ & + \int_{M^0} |\Phi[t] + \Phi_x[t] \delta x(t)| dt - \int_{M^0} |\Phi[t]| dt. \end{aligned}$$

Функционал δF_0 состоит из двух частей — дифференцируемой $\int_{M^+} - \int_{M^-}$, которая обычным образом сводится к линейному функцио-

налу от $\delta u(\cdot)$ вида $\int_0^T w_0(t) \delta u(t) dt$ (см. § 4), и интеграла по M_0 , который может быть аппроксимирован интегральной суммой по k точкам аппроксимации. После этого для вариации $\delta u(\cdot)$ мы получаем задачу: найти числа s_n из условий:

$$\begin{aligned} \text{a)} \quad & \min_{\{s_n\}} \left\{ \sum_{n=1}^N s_n h_n^0 + \sum_{j=1}^k \alpha_j \left| \Phi^{(j)} + \sum_{n=1}^N s_n h_n^{0,j} \right| \right\}; \\ \text{b)} \quad & X^i + \sum_{n=1}^N s_n h_n^i = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m; \\ \text{c)} \quad & s_n^- \leq s_n \leq s_n^+, \quad n = 1, \dots, N, \end{aligned} \quad (17)$$

где α_j — веса квадратурной формулы.

Эта задача также легко сводится к стандартной задаче линейного программирования (см. § 47).

В настоящее время автор не имеет опыта решения задач с такими функционалами (имеются, конечно, в виду задачи, в которых m с M^0 сравнима с T). Изложенное выше представляет собой некоторые предварительные соображения, но их реализация потребует дополнительных разработок. Это связано прежде всего с необходимостью создания алгоритма, использующего сравнительно небольшое число точек аппроксимации на M^0 .

§ 22. Метод поворота опорной гиперплоскости

Для решения довольно общей задачи оптимального управления:

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (1)$$

на траектории управляемой системы

$$\frac{dx}{dt} = f(x, u), \quad \Gamma(x) = 0, \quad 0 \leq t \leq T, \quad (2)$$

при условиях *)

$$\begin{aligned} F_i[u(\cdot)] &= 0, \quad i = 1, 2, \dots, m, \\ u(t) &\in U \quad \text{при } t \in [0, T] \end{aligned} \quad (3)$$

американскими математиками Итоном и Нейштадтом был предложен метод решения, использующий идеи выпуклого программирования. Сходимость метода была доказана при очень существенном предположении о строгой выпуклости области достижимости.

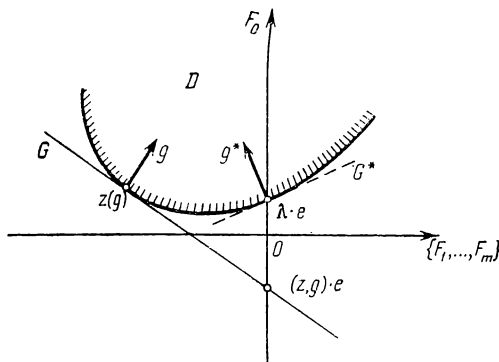


Рис. 19.

О п р е д е л е н и е. Множество точек $F[u(\cdot)] = \{F_0[u(\cdot)], \dots, F_m[u(\cdot)]\}$ в $(m+1)$ -мерном евклидовом пространстве, порожденное всеми возможными измеримыми функциями $u(t)$, определенными на $[0, T]$ и удовлетворяющими геометрическому ограничению $u(t) \in U$, называется *областью достижимости* D для задачи (1)–(3).

П р е д п о л о ж е н и е. D есть строго выпуклое ограниченное замкнутое множество.

Вариационная задача (1)–(3) эквивалентна следующей задаче выпуклого программирования: найти точку $u(\cdot)$ функционального

*) Функционалы $F_i[u(\cdot)]$ предполагаются дифференцируемыми по Фреше.

пространства управлений, отображающуюся в точку $\lambda e \in D$ с наименьшим значением λ ($e = \{1, 0, \dots, 0\}$) (рис. 19).

А л г о р и т м р е ш е н и я. Выбирается некоторый $(m+1)$ -мерный вектор $g = \{1, g_1, \dots, g_m\}$ и решается задача: найти управление $u(\cdot)$, при котором достигается

$$\min_{u(\cdot)} (F[u(\cdot)], g) = \min_{z \in D} (z, g). \quad (4)$$

По существу, это есть простейшая неклассическая задача оптимального управления без дополнительных условий, отличающаяся от исходной задачи (1)–(3) следующим:

а) минимизируемый функционал $F_0[u(\cdot)]$ заменен на функционал

$$F[u(\cdot), g] \equiv \sum_{i=0}^m g_i F_i[u(\cdot)]; \quad (5)$$

б) условий типа $F_i = 0$ в задаче нет.

Решение этой задачи может быть осуществлено простейшим вариантом метода проекции градиента (см. § 18). Разумеется, в общем случае речь идет о приближенном решении. В силу предположения о строгой выпуклости $D \min_{z \in D} (z, g)$ достигается в единственной точке $z(g) \in D$ (управление $u(\cdot)$, порождающее точку $F[u(\cdot)] = z(g)$, может и не быть единственным). Геометрический смысл величины $(z(g), g)$ легко усматривается из рис. 19: опорная к D гиперплоскость G , ортогональная вектору g и проходящая через точку $z(g) \in D$, самую низкую в D в направлении g , пересекает ось F_0 в точке $(z(g), g) e$. Таким образом, при любом векторе g , нормированном условием $(g_0, e) = 1$, имеем неравенство

$$\min_{z \in D} (z, g) \leq \lambda.$$

Целью дальнейшего является нахождение вектора g^* , определяющего гиперплоскость G^* , опорную к D в точке λe .

Этот искомый вектор g^* удовлетворяет уравнению

$$\min_{z \in D} (z, g^*) = \max_{(g, e)=1} \min_{z \in D} (z, g), \quad (6)$$

что позволяет построить сходящийся алгоритм последовательных «поворотов» опорной гиперплоскости. Именно, определим функцию

$$R(g) \equiv \min_{z \in D} (z, g), \quad g = \{1, g_1, \dots, g_m\};$$

заметив, что вычисление $R(g)$ в какой-либо точке g реализуется решением простейшей вариационной задачи с функционалом (5).

Далее решается задача поиска максимума функции $R(g)$, например, методом подъема по градиенту: имея некоторый вектор g , строим следующий g' по формуле

$$g' = g + s^* \partial R / \partial g. \quad (7)$$

Важным элементом метода является использование формулы $\partial R / \partial g = z(g)$ (теорема 42.5). Это избавляет нас от необходимости вычислять R_g численным дифференцированием, которое потребовало бы m -кратного решения вариационной задачи с функционалом (5) и дало бы не очень надежный результат, так как $R(g)$ вычисляется достаточно сложным итерационным процессом, и ошибки поиска, быть может, и не очень существенные с точки зрения величины $R(g)$, резко возрастают при численном дифференцировании R .

Что касается шага s^* , то он естественно определяется решением одномерной задачи

$$\max_s R(g + sR_g) = \max_s R[g + sz(g)]. \quad (8)$$

Она может быть решена алгоритмом параболической аппроксимации, используемым во многих расчетах, приведенных в настоящей книге. Из соответствующих таблиц в §§ 26, 27 видно, что поиск s^* обходится, примерно, в 5—6 вычислений функции $R(g)$; не надо, однако, забывать, что каждое такое вычисление — это решение простейшей вариационной задачи.

Описанный выше метод высоко ценится у теоретиков, так как хорошо вписывается в сильно развитую теорию оптимизации выпуклых функций; соответствующие теоремы обеспечивают его сходимость. Однако практическая ценность метода, видимо, невелика. Ниже мы изложим причины этого расхождения во взглядах.

1. Предположение о строгой выпуклости области достижения D существенно ограничивает область применимости метода, причем нарушение строгой выпуклости не только лишает силы доказательство сходимости, но и ликвидирует саму сходимость. В реальных нелинейных задачах проверка строгой выпуклости D фактически невозможна, предполагать же ее имеющей место нет никаких оснований. Так, в первой же задаче, в которой была численно найдена граница D (см. § 29, рис. 26), область D оказалась вогнутой.

2. Сходимость метода едва ли должна быть очень быстрой; она зависит от радиуса кривизны границы D в окрестности искомой точки l_e . Определенные трудности должны возникнуть и в связи с тем, что $R(g)$ вычисляется не очень точно алгоритмом поиска в функциональном пространстве. Стоимость одного шага (7)

в решении задачи (6) достаточно высока, так что в целом алгоритм трудоемкий.

3. Существует класс задач, где применение метода вполне обосновано; это — линейные задачи, в которых уравнения движения имеют вид

$$\frac{dx}{dt} = A(t)x + B(t)u + c(t); \quad \Gamma(x) = 0, \quad (9)$$

а функционалы $F_i[u(\cdot)]$ определяются выражениями

$$F_i[u(\cdot)] \equiv \int_0^T (\tilde{w}_i(t), u(t)) dt + \int_0^T (Y_i(t), x(t)) dt \quad (10)$$

с заданными матрицами и функциями $A, B, C, \tilde{w}, Y(t)$.

В этом случае, при выпуклой области U , область достижимости D оказывается строго выпуклой. (Это не очень точно, строгая выпуклость доказывается при некоторых предположениях, однако они выделяют общий случай, и нарушение строгой выпуклости следует считать вырождением.)

Применение метода опорной гиперплоскости в этих задачах облегчается тем, что решение вспомогательной задачи — вычисление функции $R(g)$ — осуществляется точно и довольно просто. В самом деле, выражения (10) для функционалов F_i легко приводятся к виду

$$F_i[u(\cdot)] = \int_0^T w_i(t) u(t) dt, \quad (11)$$

что требует $(m+1)$ -кратного решения задач вида

$$-\frac{d\psi}{dt} = A^*\psi + Y, \quad \Gamma^*\psi = 0. \quad (12)$$

Функции $w_i(t)$, естественно, вычисляются только один раз. Далее,

для $F[u(\cdot), g]$ имеем формулу $F[u(\cdot), g] = \int_0^T w(t) u(t) dt$, где $w(t, g) =$

$= \sum_{i=0}^m g_i w_i(t)$, и минимум $F[u(\cdot), g]$ достигается на управлении, определяемом независимыми при разных t уравнениями

$$(w(t), u(t)) = \min_{u \in U} (w(t), u). \quad (13)$$

Найдя $u(t)$ из (13) и вычислив функционалы F_i по формулам (11), получаем точку $z(g) = \{F_0, F_1, \dots, F_m[u(\cdot)]\}$.

Автору неизвестны работы, в которых сообщалось бы о фактическом использовании метода поворота опорной гиперплоскости

для решения нелинейных задач. Б. Н. Пшеничный использовал основные идеи этого метода для решения задачи быстродействия в несложной чисто методической задаче с линейной системой

$$\dot{x}^1 = x^2; \quad \dot{x}^2 = x^3; \quad \dot{x}^3 = u, \quad |u| \leq 1, \\ x(0) = 0, \quad x(T) = X$$

(строго говоря, задача быстродействия для линейной системы — задача нелинейная). Однако даже в этой простой задаче сходимость оказалась настолько медленной, что довести процесс максимизации $R(g)$ до получения удовлетворительной точности в условиях $x(T) = X$ удалось лишь после привлечения к задаче (6) идей метода сопряженных градиентов. Попробнее этот вопрос рассматривается в § 27 (см. также [65], [66]).

З а м е ч а н и е. Нелинейные задачи допускают различные эквивалентные формулировки. В частности, все входящие в постановку задачи функционалы $F_i[u(\cdot)]$ можно заменить на

$$F_i^*[u(\cdot)] = \lambda_i(F_i[u(\cdot)]) - \chi_i(0),$$

где $\chi_i(F)$ — произвольные монотонные функции (χ_0 , разумеется, монотонно растущая). Легко убедиться, что свойство выпуклости D не является инвариантным относительно подобного преобразования задачи, и это одна из причин, не позволяющая считать строгую выпуклость D «естественным» свойством реальной прикладной задачи.

Метод Нейштадта (метод поворота опорной плоскости) был предложен в [58] для задач линейного быстродействия. Сходимость его доказана, изложение доказательства можно найти, например, в [12]. Здесь мы дадим описание общей схемы метода и качественный анализ сходимости для системы

$$\frac{dx}{dt} = Ax + u; \quad u \in U, \quad x(0) = X_0. \quad (14)$$

Задача состоит в нахождении $u(\cdot)$, обеспечивающей попадание $x(T)$ в заданную точку X за кратчайшее время.

Основу метода составляет интегрирование П-системы

$$\dot{x} = Ax + u, \\ -\dot{\psi} = A^*\psi, \quad (u(t), \psi(t)) = \max_{u \in U} (u, \psi(t)). \quad (15)$$

Важную роль в дальнейшем играют следующие объекты: $D(t)$ — область достижимости за время t — это совокупность всех точек x , в которые траектория системы (14) может попасть под воздействием какого-то управления за время $t' \leq t$; $G(t) = \partial_t D(t)$ — граница $D(t^*)$.

*) Точнее, $\partial_t D(t) = \lim_{dt \rightarrow +0} D(t + dt)/D(t)$.

Будем здесь предполагать, что $D(t)$ — строго выпуклая область (для широкого класса линейных задач эта строгая выпуклость довольно просто доказывается; см., например, [12]). Из теории линейных задач известно, что решение Π -системы $\{x(t), \phi(t)\}$ обладает следующим свойством: $x(t) \in G(t)$, а $\phi(t)$ определяет опорную к $G(t)$ в точке $x(t)$ гиперплоскость; если в этой точке поверхность $G(t)$ дифференцируема, то $\phi(t)$ есть внешняя нормаль к ней.

Решение задачи начинается заданием какого-то $\phi(0)$, стесненного лишь требованием $(X - X_0, \phi(0)) > 0$ (например, $\phi(0) = X - X_0$). С этим значением $\phi(0)$ Π -система интегрируется до момента t^* , определяемого условием

$$(\phi(t^*), X - x(t^*)) = 0. \quad (16)$$

Такой момент наступит, так как при достаточно большом t $X \in D(t)$, следовательно,

$$(\phi(t), X - x(t)) \leq 0.$$

Кроме того, $\phi(t)$ и $x(t)$ — непрерывны.

Итак, уравнением (16), а также процессом интегрирования Π -системы устанавливается функциональная зависимость $t^*(\phi_0)$. Если в момент $t_0^* = t^*(\phi_0)$ будет $x(t_0^*) = X$, то задача решена.

Очевидно, что наименьшее время перехода $x(t)$ из X_0 в X есть то минимальное время T , при котором расширяющееся с ростом t множество достижимости $D(t)$ впервые поглощает точку X , т. е. время оптимального быстрогодействия T есть наименьший «корень» уравнения

$$X \in D(t) \quad (\text{или } X \in G(t)).$$

Если при заданном ϕ_0 в момент $t^*(\phi_0) = t_0^*$ окажется, что $x(t_0^*) \neq X$, то в силу строгой выпуклости $D(t^*)$ X лежит вне D и, следовательно, $t^*(\phi_0) \leq T$ при всех ϕ_0 . С другой стороны в силу принципа максимума существует решение Π -системы, соединяющее X_0 с X за время T . Итак, доказана

Л е м м а 1. *Искомые начальные данные $\phi(0)$ удовлетворяют уравнению*

$$\max_{\phi(0)} t^*[\phi(0)] = T.$$

Таким образом, задача быстрогодействия свелась к поиску $\max_{\phi_0} t^*(\phi_0)$.

Учитывая сложный неявный характер определения зависимости $t^*(\phi_0)$, мы можем, в общем случае, рассчитывать лишь на численные методы поиска максимума, для чего полезно уметь вычислять $\partial t^*(\phi_0)/\partial \phi_0$. Вычисление этой производной является хорошим упражнением, но нам она, в сущности, не понадобится.

Прежде чем двигаться дальше, проиллюстрируем сказанное выше на примере простой задачи быстрого действия, решение которой хорошо известно:

$$\dot{x}^1 = x^2; \quad \dot{x}^2 = u, \quad |u| \leq 1, \quad x(0) = 0.$$

На рис. 20 качественно изображены введенные выше объекты: область достижимости $D(t^*)$, ее граница $G(t^*)$, траектория $x(t)$, определяемая решением П-системы с начальным вектором ψ_0 ,

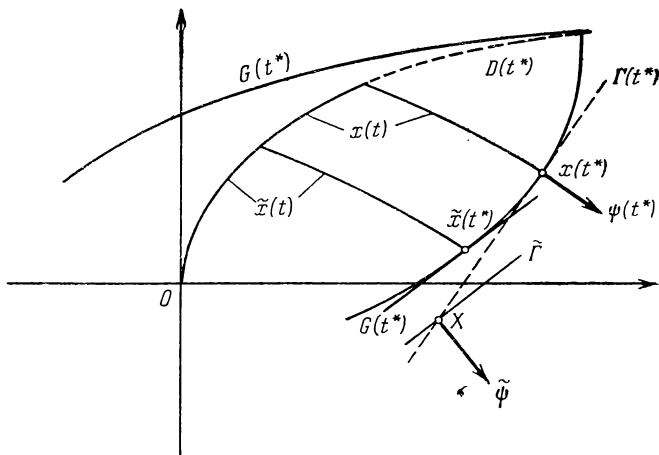


Рис. 20.

$\phi(t^*)$ есть нормаль к $G(t^*)$, а t_0^* определяется как первый момент, когда гиперплоскость $\Gamma(t)$, проведенная через $x(t)$ ортогонально $\phi(t)$, пройдет через заданное конечное состояние X .

Теперь повернем слегка гиперплоскость Γ около точки X , определив ее нормалью

$$\tilde{\phi} = \phi(t^*) + \alpha [X - x(t^*)]. \quad (17)$$

Может быть сформулирована

Л е м м а 2. При достаточно малом $\alpha > 0$ область $D(t^*)$ лежит «ниже» гиперплоскости $\tilde{\Gamma}$, т. е.

$$(\tilde{\phi}, x - X) < 0 \quad \text{для всех } x \in D(t^*).$$

Доказательство предоставим читателю.

Построим теперь новую траекторию $\tilde{x}(t)$ так, чтобы она удовлетворяла принципу максимума и в момент t^* выходила на ту точку границы $G(t^*)$, в которой $\tilde{\phi}$ является нормалью (см. рис. 20). Построить такую траекторию несложно: нужно проинтегрировать уравнение $-\dot{\phi} = A^* \phi$ назад, от t^* к 0, задав данные Коши $\phi(t^*) = \tilde{\phi}$;

имея $\psi(t)$, из принципа максимума находим $u(t)$ на интервале $[0, t^*]$ и соответствующую фазовую траекторию $\tilde{x}(t)$. Очевидно следующее:

С л е д с т в и е. Гиперплоскость, проведенная через точку $\tilde{x}(t^*)$ ортогонально вектору ψ , строго отделяет $D(t^*)$ от X ; другими словами,

$$(\tilde{\psi}, X - \tilde{x}(t^*)) > 0.$$

Теперь можно продолжить траекторию $\tilde{x}(t)$ для $t > t^*$, решая совместно II-систему с начальными данными при t^* ; в некоторый момент $t_1^* > t^*$ реализуется ситуация

$$(\tilde{\psi}(t_1^*), X - \tilde{x}(t_1^*)) = 0.$$

Этим и заканчивается стандартный шаг процесса максимизации $t^*(\psi_0)$. Как уже говорилось, сходимость $t^* \rightarrow T$ доказана; одновременно с построением максимизирующей последовательности $t_0^* < t_1^* < t_2^* \dots \rightarrow T$ мы получаем и последовательность оптимальных (удовлетворяющих принципу максимума) траекторий, правые концы которых $x(t_0^*), x(t_1^*), \dots \rightarrow X$, а сами траектории сходятся к искомой. Подчеркнем еще раз, что существенным фактором, определяющим успех этого метода, является строгая выпуклость множеств достижимости $D(t)$. Поскольку для области $D(t)$, выпуклой, но не строго, метод может не сходиться, следует ожидать медленной сходимости в тех ситуациях, когда граница $G(t)$ в окрестности X имеет малую кривизну.

Практическая реализация алгоритма требует решения еще одного важного вопроса: как выбирать «шаг» α в формуле (17)? В расчетах Б. Н. Пшеничного и Л. А. Соболенко, результаты которых мы кратко обсудим, выбор α осуществлялся так: описанная выше процедура для каждого α дает значение $t^*(\alpha)$, удовлетворяющее уравнению типа (16). Вычисляя $t^*(\alpha)$ для нескольких значений, например, $0, h, 2h$, и используя далее квадратичную интерполяцию $t^*(\alpha)$, можно с необходимой точностью найти $\max t^*(\alpha)$;

точка максимума α^* и определяет шаг процесса. В [67] приведены результаты вычислительного эксперимента для системы $\dot{x}^1 = x^2$, $\dot{x}^2 = x^3$, $\dot{x}^3 = u$, $|u| \leq 1$, $x(0) = 0$, $X = \{a, 0, 0\}$:

1) при $a=2$ для попадания в X с точностью $0, 1$ (т. е. $\|x(t^*) - X\| \sim 0,1$) потребовалось 37 итераций (в статье, к сожалению, не указывается, сколько интегрирований II-системы составляют эти 37 итераций);

2) в том же примере для достижения точности $0,01$ потребовалось 80 итераций (видно, что уточнение решения в методе Нейштатда происходит плохо);

3) задача с $a=16$ не была решена за разумное машинное время (при точности $0,1$); это связано, конечно, с тем, что граница

$G(T)$ в этом случае имеет заметно меньшую кривизну, чем при $a=2$ ($T=4$).

Используя для решения задачи поиска $\max_{\phi_0} t^*(\phi_0)$ описанную в § 51 процедуру метода сопряженных градиентов, задачи 1), 2), 3) удалось решить за 6, 10 и 13 итераций соответственно.

Сравнивая метод Нейштадта (метод поворота опорной гиперплоскости) с методом Ньютона*), можно сделать следующие выводы.

1. При относительно хорошем начальном приближении $\{\phi_0, T\}$ метод Ньютона имеет заметное преимущество, быстро приводя к практически точному решению. Метод Нейштадта в этой ситуации, даже усиленный идеями метода сопряженных градиентов, сходится довольно медленно.

2. Однако на стадии поиска, начинающейся с грубого приближения ϕ_0 , метод Нейштадта, видимо, эффективнее метода Ньютона: в частности, если начальное значение ϕ_0 дает $\phi(t)$, имеющую меньше, чем нужно, нулей, метод Ньютона встречает затруднения. Для метода поворота опорной гиперплоскости подобных затруднений не возникает. Сделать более уверенные выводы, к сожалению, не удастся, так как в [65], [66] результаты экспериментов приведены очень скупо: нет, в частности, указаний о выборе начальных данных, о ходе итерационного процесса.

§ 23. Приближенное решение задач со скользящим режимом

Рассмотрим простую задачу, в решении которой появляется скользящий режим. Эта задача уже была проанализирована в § 10, здесь же мы попробуем разобраться в том, какие осложнения возникнут при попытке ее численного решения. Итак, рассматривается модельная управляемая система:

$$\begin{aligned} \dot{x}^1 &= x^2 + [u(t)]^2; & x^1(0) &= 0, \\ \dot{x}^2 &= u(t); & x^2(0) &= 0; & 0 \leq t \leq T = 3. \end{aligned} \quad (1)$$

Задача состоит в определении $u(t)$ из условий

$$\begin{aligned} \max_{u(\cdot)} x^1(T), \\ x^2(T) = 0; \quad |u(t)| \leq 1; \quad x^2(t) \leq 1. \end{aligned} \quad (2)$$

Попытаемся решать ее численно методом последовательной линей-

*) См. § 27.

ризации, взяв в качестве исходного приближения изображенную на рис. 21 функцию. Ей соответствует управление

$$u(t) = \begin{cases} 1, & 0 \leq t < 1, \\ -1, & 1 \leq t < 1,5, \\ 1, & 1,5 \leq t < 2, \\ -1, & 2 \leq t \leq 3. \end{cases} \quad (3)$$

Опуская несложные выкладки, найдем выражения для вариаций функционалов

$$\delta x^1(T) = \int_0^T [2u(t) + (T-t)] \delta u(t) dt = \int_0^T w_0 \delta u dt,$$

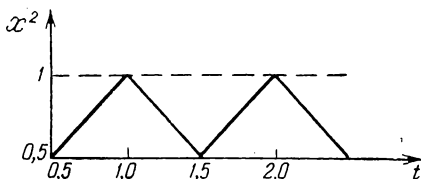
$$\delta x^2(T) = \int_0^T \delta u(t) dt,$$

$$\delta x^2(1) = \int_0^1 \delta u(t) dt; \quad \delta x^2(2) = \int_0^2 \delta u(t) dt.$$

Будем решать задачу в классе кусочно постоянных управлений

$$u(t) = u_{n+1/2} \quad \text{при} \quad n\tau \leq t < (n+1)\tau; \quad \tau = T/N.$$

Если N достаточно велико, мы можем в этом классе получить очень точную аппроксимацию скользящего режима: на интервале $1 < t < 2$ приближенная величина $x^2(t)$ будет отличаться от точной (соответствующей скользящему режиму) $x^2(t) = 1$ на величину, не большую τ . Однако попытка численного решения задачи сразу же оказывается безуспешной: дело в том, что управление (3) является точкой (в функциональном пространстве) локального максимума, достаточно далекой от точки глобального максимума. Следует сразу же разъяснить: это управление является точкой локального максимума лишь относительно класса малых вариаций управления. Относительно класса конечных вариаций управления на множестве малой меры оно точкой локального максимума не будет. В этом читатель без труда убедится, если проварьирует управление лишь на двух малых интервалах сетки, примыкающих к точке $t = 1,5$; на левом следует заменить $u(t) = -1$ на $u = +1$,



[Рис. 21.

а на правом — наоборот. Будет получена траектория, не нарушающая ни одного из дополнительных условий, а значение функционала

$$x^1(T) = \int_0^T \{x^2(t) + [u(t)]^2\} dt$$

увеличится, так как $x^2(t)$ возрастает. Немногим более сложна проверка того, что в классе малых вариаций управления точка (3) — неулучшаема. В самом деле, из (3) следует очевидная структура конуса вариаций управления:

$$\begin{aligned} \delta u(t) &\leq 0 \quad \text{при } t \in (0; 1) \cup (1,5; 2), \\ \delta u(t) &\geq 0 \quad \text{при } t \in (1; 1,5) \cup (2; 3). \end{aligned} \quad (4)$$

В этих условиях в классе функций (4) не существует решения задачи

$$\delta x^1(T) = \int_0^T w_0(t) \delta u(t) dt > 0 \quad (5)$$

(мы даже не учитываем дополнительных ограничений $\int_0^T \delta u dt = 0$, $\int_0^1 \delta u dt = 0$, $\int_0^2 \delta u dt = 0$). В самом деле, заметим, что

$$\begin{aligned} w_0(t) &= 2u(t) + (T - t) > 0 \quad \text{при } t \in (0, 1) \cup (1, 5; 2), \\ w_0(t) &< 0 \quad \text{при } t \in (1; 1, 5) \cup (2; 3). \end{aligned}$$

Так же можно убедиться, что для малых вариаций $\delta u(t)$ точкой локального максимума будет любое управление, равное ± 1 на чередующейся последовательности интервалов. Таким образом, на пути к достаточно точной аппроксимации скользящего режима алгоритм приближенного решения, основанный на малых вариациях $\delta u(t)$, встретит огромное число локальных экстремумов, в каждом из которых процесс может «застрять». Эта ситуация характерна для задач со скользящими режимами. Преодолеть такие трудности можно с помощью алгоритмов, в которых минимизирующая последовательность управлений строится процессом конечных вариаций управления на множестве малой меры. В данном примере легко реализовать такой процесс и продемонстрировать его эффективность. Однако эта легкость была бы следствием тривиальности самой задачи: ведь она без труда решается «в уме».

Посмотрим теперь с какими проблемами мы встретимся при попытке в сравнительно общей задаче осуществить построение конечной вариации на множестве малой меры. Напомним (см. § 6), что в этом случае для вариаций функционалов задачи $F_i[u(\cdot)]$ имеются формулы

$$\delta F_i = \int_0^T \psi^{(i)}(t) \{f[x(t), v(t)] - f[x(t), u(t)]\} dt \quad (6)$$

(здесь $\{x(t), u(t)\}$ — невозмущенная траектория, $\psi^{(i)}(t)$ — определенные решения сопряженного уравнения, $v(t)$ — управляющая функция, отличающаяся от $u(t)$ лишь на множестве малой меры, а в остальном произвольная). Таким образом, естественная задача, определяющая вариацию управления, в этом случае формулируется так: задано малое $\varepsilon > 0$, следует найти множество $M \subset [0, T]$ и функцию $v(t)$ из условий:

- 1) $\min_{M, v(\cdot)} \int_M \psi^{(0)} \{f(x, v) - f(x, u)\} dt,$
- 2) $F_i + \int_M \psi^{(i)} \{f(x, v) - f(x, u)\} dt,$ (7)
- 3) $\text{mes } M \leq \varepsilon, v(t) \in U.$

Это — достаточно сложная задача, как ее решать — не совсем ясно. Можно несколько упростить ее, требуя не минимизации δF_0 , а лишь выполнения неравенства $\delta F_0[M, v(\cdot)] < 0$. Это, конечно, сделает процесс построения минимизирующей последовательности менее эффективным (замедлится скорость сходимости к экстремуму). Можно избрать и промежуточный вариант, выбирая множество M из каких-то «разумных» соображений, а для определения $v(\cdot)$ решать все-таки задачу на условный экстремум. Но и после этого задача остается сложной, а ведь ее предстоит решать многократно. К тому же, работая в условиях невыпуклой области $f(x, U)$, мы можем столкнуться с проблемой нелокального экстремума. Таким образом, этот подход реализуем, видимо, лишь в двух ситуациях:

- 1) если аналитические выражения $f(x, u)$ настолько просты, что задачу типа (7) можно решить частными приемами;
- 2) при отсутствии дополнительных условий (7.2), когда задача (7) является задачей на безусловный минимум.

Не нужно только последний случай трактовать слишком широко, сводя к нему с помощью штрафных функций самую общую задачу. Дело ведь не только в словесном оформлении задачи —

задача на условный экстремум очень сложна, а задача на безусловный экстремум много проще. Очень важными факторами являются свойства гладкости участвующих в постановке задачи функций. А метод штрафных функций основан на введении в задачу очень негладких функций, после чего и задача на безусловный экстремум становится очень сложной.

По мнению автора, наиболее реальным подходом к решению задач со скользящими режимами является прием, использованный в теоретических целях Р. В. Гамкрелидзе (см. стр. 88); Правда, и здесь не все так уж просто, так как размерность управления существенно повышается. Напомним, что метод Гамкрелидзе состоит в замене управляемой системы $\dot{x}=f(x, u)$ системой

$$\dot{x} = \sum_{i=1}^{p+1} \alpha_i(t) f(x, u_i),$$

причем вместо одной вектор-функции $u(t)$ в исходном уравнении появляются p управляющих вектор-функций $u_i(t)$ плюс p скалярных управлений $\alpha_i(t)$ (число p связано с размерностью тела $\text{conv } f(x, U)$). Во всяком случае ясно, что решение задач со скользящими режимами не очень просто, здесь еще нет опыта, и браться за это дело стоит только в том случае, когда это действительно необходимо.

В § 10 при обсуждении вопроса о том, как часто в прикладных задачах могут появляться скользящие режимы, автор утверждал, что, например, в задачах § 30 с невыпуклой вектограммой, где в принципе возможны скользящие режимы, содержательные постановки задач к таким решениям не приводят. Правда, эти задачи решались автором методом малых вариаций управлений. Такой метод и не рассчитан на получение скользящих режимов. Тем не менее, можно с достаточными основаниями утверждать, что не в этом дело. Прежде всего, во всех этих задачах почти очевидно, что уменьшение сопротивления воздуха движению тела является благоприятным фактором с точки зрения поставленной вариационной задачи. Поэтому режим, в котором угол атаки ($u_2(t)$) в среднем близок к нулю, а его квадрат (сопротивление воздуха пропорционально u_2^2) — велик, едва ли может быть оптимальным. Другим соображением являются результаты проводившихся автором экспериментов по решению задачи (1) методом малых вариаций управления (§§ 19—21).

Начиная с каких-то траекторий, алгоритм быстро приводил к локальному экстремуму — траектории, качественно близкой к изображенной на рис. 21. Эти локально экстремальные траектории в зависимости от начального управления имели существенно разную форму, разные значения функционала F_0 , но все они состояли из релейных управлений $u(t) = +1 (-1)$ на чередующейся

последовательности отрезков. Таким образом, численные результаты содержали явное указание на скользящий режим, понятное, разумеется, тому, кто знаком с этим явлением. В задачах § 30 решения, начинающиеся с разных начальных управлений, приводят практически к одной и той же траектории $x(t)$, к одному и тому же значению F_0 , хотя $u(t)$ могут отличаться достаточно сильно. Такая картина связана прежде всего с некорректностью задачи (см. § 40) и не дает оснований подозревать наличие скользящего режима (скрытого решения).

§ 24. Градиентный метод второго порядка

В §§ 18—23 были описаны методы построения минимизирующей последовательности управлений, использующие лишь первые производные входящих в задачу функционалов. Поэтому эти методы называют *методами первого порядка*. Давно было замечено, что при решении задач поиска минимума методом первого порядка сходимость оказывается очень медленной в окрестности точки минимума. Это и понятно: ведь в этой окрестности, грубо говоря, первая производная минимизируемого функционала обращается в нуль, и приращение его при вариации аргумента (управления) определяется вторым членом разложения. Стремясь повысить скорость поиска и получить более точные результаты без существенного увеличения времени счета, естественно приходят к идее использования в вычислениях также вторых производных от функционалов задачи. Кроме того, с этим же связаны и надежды повысить эффективность поиска в условиях применения штрафных функций, когда сходимость методов первого порядка оказывается очень медленной даже сравнительно далеко от искомой точки минимума. *Методы второго порядка* разработаны не так подробно, как методы первого порядка, а опыт их фактического применения совсем невелик. Ниже будет описана общая схема метода второго порядка и рассмотрены возникающие при его реализации вычислительные проблемы.

Итак, рассматривается управляемая система

$$\frac{dx}{dt} = f(x, u), \quad x(0) = X_0 \quad 0 \leq t \leq T, \quad (1)$$

где X_0 , T заданы. На траектории системы определены функционалы вида

$$F_i[u(\cdot)] \equiv \int_0^T \Phi^i[x(t), u(t)] dt + R^i[x(T)], \quad i = 0, 1, \dots, m. \quad (2)$$

Ищется

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (3)$$

при условиях

$$F_i[u(\cdot)] = 0, \quad i = 1, 2, \dots, m; \quad u(t) \in U. \quad (4)$$

Пусть $\{u(\cdot), x(\cdot)\}$ — некоторая «невозмущенная» траектория системы. Пусть управление возмущено малой функцией $\delta u(\cdot)$, следствием чего явилось малое возмущение фазовой траектории: $x(t) \rightarrow x(t) + \delta x(t)$.

Уравнение в вариациях определяет связь между $\delta x(t)$ и $\delta u(t)$, однако в теории второго порядка оно имеет такую форму:

$$\begin{aligned} \frac{d\delta x}{dt} = f_x \delta x + f_u \delta u + \frac{1}{2} (f_{xx} \delta x, \delta x) + \\ + \frac{1}{2} (f_{uu} \delta u, \delta u) + (f_{xu} \delta x, \delta u), \quad \delta x(0) = 0. \end{aligned} \quad (5)$$

Формально уравнением (5) функция $\delta x(t)$ однозначно определяется при заданном $\delta u(\cdot)$, однако это уравнение — нелинейное, и вся теория приобретает существенно более сложный характер. В то же время нужно иметь в виду и важное благоприятное обстоятельство: поскольку δx и δu являются малыми величинами, решение уравнения (5) может быть осуществлено методом итераций: сначала находится $\delta \tilde{x}(t)$ решением линейного уравнения

$$\frac{d\delta \tilde{x}}{dt} = f_x \delta \tilde{x} + f_u \delta u; \quad \delta \tilde{x}(0) = 0, \quad (5^*)$$

затем в квадратичные члены уравнения (5) подставляется это первое приближение $\delta \tilde{x}$, и следующее приближение $\delta x(t)$ находится решением линейной системы

$$\begin{aligned} \frac{d\delta x}{dt} - f_x[t] \delta x = f_u[t] \delta u + \frac{1}{2} (f_{xx} \delta \tilde{x}, \delta \tilde{x}) + \\ + \frac{1}{2} (f_{uu} \delta u, \delta u) + (f_{xu} \delta \tilde{x}, \delta u), \quad \delta x(0) = 0. \end{aligned} \quad (5^{**})$$

Для наших целей этого достаточно, и дальнейших итераций можно будет не проводить, поскольку они дают поправки к δx , имеющие формально третий порядок малости. Разумеется, как всегда, мы считаем, что функции $f(x, u)$, $\Phi(x, u)$, $R(x)$ имеют непрерывные производные нужных порядков; f_x, f_u, f_{xx} и т. д. — суть функции t (векторные и матричные), определенные на невозмущенной траектории $\{u(t), x(t)\}$.

Заметим, что мы не случайно ограничиваемся здесь функционалами вида (2). Для функционалов

$$F[u(\cdot)] \equiv \max_t \Phi[x(t)] \quad (2^*)$$

существование первой производной по направлению гарантируется достаточной гладкостью функции $\Phi(x)$ и является естественным

в прикладных задачах, в то время как существование второй производной по направлению уже не обеспечивается предположением какой угодно гладкости Φ . Это связано с тем, что формула для приращения функционала (2*), установленная в § 4, имеет вид

$$F[u(\cdot) + \varepsilon v(\cdot)] = F[u(\cdot)] + \varepsilon D[u(\cdot), v(\cdot)] + o(\varepsilon),$$

и заменить $o(\varepsilon)$ на $O(\varepsilon^2)$ в общем случае нельзя.

Прямая вариация функционала дает следующую простую формулу

$$\begin{aligned} \delta F[\delta u(\cdot)] = & \int_0^T \Phi_x \delta x(t) dt + \int_0^T \Phi_u \delta u dt + R_x \delta x(T) + \\ & + \frac{1}{2} \int_0^T (\Phi_{xx} \delta x, \delta x) dt + \frac{1}{2} \int_0^T (\Phi_{uu} \delta u, \delta u) dt + \\ & + \int_0^T (\Phi_{ux} \delta u, \delta x) dt + \frac{1}{2} (R_{xx} \delta x(T), \delta x(T)). \end{aligned} \quad (6)$$

Теперь, используя уравнение в вариациях (5), следует в выражении (6) исключить вариацию зависимого аргумента $\delta x(t)$ через $\delta u(\cdot)$ и получить $\delta F[\delta u(\cdot)]$ в виде квадратичного функционала от $\delta u(\cdot)$. Здесь мы также можем воспользоваться сделанным выше замечанием и в квадратичных членах использовать связь между $\delta x(t)$ и $\delta u(\cdot)$, следующую из линейной теории (5*). Эта связь нам будет нужна и для вычислений, поэтому выпишем ее в явном виде:

$$\delta x(t) = \int_0^t G(t, \tau) \delta u(\tau) d\tau. \quad (7)$$

Функция $G(t, \tau)$ связана с функцией Грина линейной задачи (5*); ее вычисление сводится к следующим операциям:

1. Найти решение матричного уравнения в вариациях:

$$\frac{d\mathcal{E}}{dt} - f_x[t] \mathcal{E}(t) = 0; \quad \mathcal{E}(0) = E, \quad (8)$$

что эквивалентно n -кратному интегрированию системы линейных уравнений (5*) (n — размерность x). $\mathcal{E}(t)$ — матрица $n \rightarrow n$.

2. $G(t, \tau) = \mathcal{E}(t) \mathcal{E}^{-1}(\tau) f_u[\tau].$

Таким образом, $G(t, \tau)$ есть матрица $r \rightarrow n$ (r — размерность u). Матрица $G(t, \tau)$ нужна для дальнейших вычислений и ее следует «запомнить» в виде таблицы значений на временной сетке с N

узлами. Это потребует N^2rn ячеек памяти (в общем случае; в конкретной задаче необходимый ресурс памяти может быть меньшим)*).

Теперь выражение (7) может быть подставлено в квадратичные члены формулы (6), превращая последние в квадратичные функционалы от $\delta u(\cdot)$. Подробно поясним необходимые вычисления на самом сложном выражении

$$\begin{aligned} \int_0^T (\Phi_{xx}[t] \delta x(t), \delta x(t)) dt &= \\ &= \int_0^T (\Phi_{xx}[t] \int_0^t G(t, \tau) \delta u(\tau) d\tau, \int_0^t G(t, \xi) \delta u(\xi) d\xi) dt = \\ &= \int_0^T \int_0^T (A_1(\xi, \tau) \delta u(\tau), \delta u(\xi)) d\tau d\xi, \end{aligned}$$

где матрица $r \rightarrow r$

$$A_1(\xi, \tau) = \int_{\max(\xi, \tau)}^T G^*(t, \xi) \Phi_{xx}[t] G(t, \tau) dt.$$

Что касается линейных по $\delta x(t)$ членов в (6), т. е. $\int_0^T \Phi_x \delta x dt$ и $R_x \delta x(T)$, то они должны быть преобразованы с использованием уравнения в вариациях вида (5**), в которое вместо $\delta \tilde{x}(t)$ подставлено выражение (7). В этом случае уравнение (5**) принимает вид

$$\delta \tilde{x} - f_x \delta x = f_u \delta u(t) + \int_0^t \int_0^t (A_2(t, \tau, \xi) \delta u(\tau), \delta u(\xi)) d\tau d\xi, \quad (9)$$

где $A_2(t, \tau, \xi)$ — тензор такого типа, что выражение $(A_2(t, \tau, \xi) \delta u, \delta u')$ есть вектор размерности n^{**}). Формулы для вычисления его компонент получаются из выражения

$$\begin{aligned} \frac{1}{2} \left(f_{xx}[t] \int_0^t G(t, \tau) \delta u(\tau) d\tau, \int_0^t G(t, \xi) \delta u(\xi) d\xi \right) + \\ + \left(f_{ux}[t] \delta u(t), \int_0^t G(t, \tau) \delta u(\tau) d\tau \right) + \frac{1}{2} \left(f_{uu}[t] \delta u(t), \delta u(t) \right) \end{aligned}$$

*) Впрочем, можно запомнить на сетке лишь матрицы $\mathcal{E}(t)$, $\mathcal{E}^{-1}(t)$, $f_u[t]$, что потребует $2Nn^2 + Nrn$ ячеек.

**) Можно представлять A_2 в виде n -вектора, компоненты которого суть матрицы $r \rightarrow r$, зависящие от трех переменных t, τ, ξ .

приведением его выкладками, аналогичными проделанным выше, к виду

$$\int_0^t \int_0^\xi (A_2(t, \tau, \xi) \delta u(\tau), \delta u(\xi)) d\tau d\xi.$$

Далее решение уравнения (9) записывается в виде

$$\delta x(t) = \int_0^t G(t, \tau) \delta u(\tau) d\tau + \int_0^t \int_0^\eta \int_0^\xi (A_2(\eta, \tau, \xi) \delta u(\tau), \delta u(\xi)) d\tau d\eta d\xi;$$

можно записать его и в более удобной для дальнейшего форме

$$\delta x(t) = \int_0^t G(t, \tau) \delta u(\tau) d\tau + \int_0^t \int_0^\tau (\bar{A}_2(t, \tau, \xi) \delta u(\tau), \delta u(\xi)) d\tau d\xi, \quad (10)$$

проделав соответствующие преобразования. Затем это выражение следует подставить в те части формулы (6), которые линейны по $\delta x(t)$. После некоторых преобразований формула (6) приобретет следующий вид:

$$\delta F[\delta u(\cdot)] = \int_0^T w(t) \delta u(t) dt + \int_0^T \int_0^T (W(\xi, \tau) \delta u(\xi), \delta u(\tau)) d\xi d\tau, \quad (6^*)$$

где W — матрица $r \rightarrow r$. Выше мы намеренно не доводили всех выкладок до конца, ограничиваясь лишь указанием той формы, к которой можно и нужно привести те или иные выражения. Для дальнейшего нам важны следующие выводы.

1. Можно написать выражение для $\delta F[\delta u(\cdot)]$, имеющее точность $O(\|\delta u\|^3)$, в виде квадратичного функционала от $\delta u(\cdot)$.

2. Вычисление матрицы $W(\xi, \tau)$ сводится к последовательным квадратурам от известных, определенных на невозмущенной траектории, функций.

3. Используя в вычислениях кусочно постоянные сеточные функции $u(t)$ на сетке с N интервалами, превратим квадратичный функционал в квадратичную форму в пространстве размерности Nr . Для того чтобы работать в дальнейшем с подобными формами, следует запомнить на сетке матрицу-функцию $W(\xi, \tau)$, что потребует $N^2 r^2$ ячеек памяти. Для того чтобы использовать формулы второго порядка точности для всех $(m+1)$ функционалов вариационной задачи, потребуется $(m+1)N^2 r^2$ ячеек памяти. Допустим, однако, что затруднения с памятью оказались преодолимыми, и все вектор-функции $w_i(t)$ и матрицы-функции $W_i(\xi, \tau)$ ($i=0, 1, \dots, m$) вычислены и хранятся в памяти в виде соответствующих таблиц для дискретных значений аргументов t, ξ, τ .

Теперь для определения вариации $\delta u(\cdot)$ может быть поставлена следующая задача:

$$\min_{\delta u(\cdot)} \left\{ \int_0^T w_0(t) \delta u(t) dt + \int_0^T \int_0^T (W_0(\tau, \xi) \delta u(\tau), \delta u(\xi)) d\tau d\xi \right\}, \quad (11)$$

при условиях

$$F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt + \int_0^T \int_0^T (W_i(\xi, \tau) \delta u(\xi), \delta u(\tau)) d\xi d\tau = 0, \quad (12)$$

$$i = 1, 2, \dots, m, \quad (13)$$

(ради простоты мы не делаем очевидной замены интегралов конечными суммами).

Формально получена сложная задача, однако и здесь настраивается итерационный метод ее решения, при котором все условия (12) берутся в линейной форме, а квадратичные члены берутся из предыдущей итерации. Таким образом, каждый шаг этой процедуры потребует решения задачи на минимум квадратичного функционала при линейных ограничениях, что уже значительно проще: соответствующие алгоритмы описаны, например, в § 51. Мы ограничимся этим беглым и общим описанием, потому что в такой форме методы второго порядка, учитывая всю громоздкость предварительных вычислений, в сложных задачах применять будет, видимо, очень трудно и едва ли рационально. Однако можно ввести некоторые упрощения и получить более практичные, хотя и не столь последовательные, методы.

Практическая форма метода основывается на следующих упрощениях.

1. Уравнение в вариациях используется в линейной форме

$$\delta \dot{x} = f_x[t] \delta x + f_u[t] \delta u, \quad \delta x(0) = 0,$$

а его решение в виде (7).

2. Вариации всех функционалов $F_1, \dots, F_m[u(\cdot)]$ также вычисляются лишь в линейном приближении

$$\delta F_i[\delta u(\cdot)] = \int_0^T \Phi_u^i \delta u dt + \int_0^T \Phi_x^i \delta x dt + R_x^i \delta x(T).$$

3. Вариация минимизируемого функционала вычисляется с учетом квадратичных членов, т. е. в виде (11). Обычным образом

для $\delta F_i [\delta u(\cdot)]$ получаем выражение

$$\delta F_i [\delta u(\cdot)] = \int_0^T w_i(t) \delta u(t) dt, \quad i = 1, 2, \dots, m,$$

а для $\delta F_0 [\delta u(\cdot)]$ — полное выражение

$$\delta F_0 [\delta u(\cdot)] = \int_0^T w_0(t) \delta u(t) dt + \int_0^T \int_0^T (W_0(\xi, \tau) \delta u(\xi), \delta u(\tau)) d\xi d\tau,$$

причем вычисление функции $W_0(\xi, \tau)$ существенно упрощается в силу более простой формы уравнения в вариациях.

Разумеется, теперь нельзя говорить о методе второго порядка, однако можно привести соображения в пользу такого непосредственного подхода: ведь в окрестности минимума вырождается (обращается в нуль) линейная часть приращения δF_0 , поэтому естественно уточнить вычисления именно в этом месте *). Учитывая в условиях $F_i = 0$, $i = 1, 2, \dots, m$, лишь линейные по $\delta u(\cdot)$ члены, мы будем получать невязки $F_i [u(\cdot) + \delta u(\cdot)] \simeq O(\|\delta u\|^2)$, и их компенсация на следующей итерации потребует малой, порядка $\|\delta u\|^2$, части вариации управления. Труднее оправдать использование простейшей формы уравнения в вариациях при преобразовании линейной по $\delta x(t)$ части вариации функционала F_0 . Видимо, решающим аргументом здесь является относительная простота преобразования исходного выражения для δF_0 (6). Выше мы убедились в том, что, используя линейную связь между $\delta x(t)$ и $\delta u(\cdot)$, нетрудно довести выкладки до конца и преобразовать первоначальное выражение δF_0 (в виде квадратичной формы от $\delta x(\cdot)$ и $\delta u(\cdot)$) в квадратичную форму только от $\delta u(\cdot)$. Попытка проделать ту же операцию, используя более точную форму уравнения в вариациях, хотя и не встретила принципиальных трудностей, однако привела к существенному усложнению всей процедуры, так что ее не так просто довести до конца даже на уровне формальных выкладок.

Так или иначе, мы приходим к следующей задаче определения вариации управления:

$$\min_{\delta u(\cdot)} \left\{ \int_0^T w_0(t) \delta u(t) dt + \frac{1}{2} \int_0^T \int_0^T (W_0(\xi, \tau) \delta u(\xi), \delta u(\tau)) d\xi d\tau \right\}, \quad (14)$$

*) Стоит уточнить, что означает это «вырождение» в окрестности минимума. Если в задаче нет условий $F_i = 0$, то в тех точках t , где $u(t)$ не вышло на границы U , $w_0(t) \simeq 0$. Если же условия $F_i = 0$ в задаче есть,

то для всех $\delta u(\cdot)$, удовлетворяющих условиям (12), $\int_0^T w_0 \delta u dt \geq C \|\delta u\|^2$.

при условиях

$$F_i[u(\cdot)] + \int_0^T w_i(t) \delta u(t) dt = 0, \quad i = 1, 2, \dots, m, \\ u(t) + \delta u(t) \in U \quad (15)$$

или

$$\int_0^T \|\delta u(t)\|^2 dt = S^2. \quad (16)$$

Используя класс кусочно постоянных управляющих функций на сетке с N малыми временными интервалами, покрывающими $[0, T]$, приходим к следующей задаче квадратичного программирования в конечномерном пространстве:

$$\min_{\{s_n\}} \left\{ \sum_{n=1}^N s_n h_n^0 + \frac{1}{2} \sum_{p=1}^N \sum_{q=1}^N H_{pq}^0 s_p s_q \right\}, \quad (14^*)$$

при условиях

$$X^i + \sum_{n=1}^N h_n^i s_n = 0, \quad i = 1, 2, \dots, m, \quad (15^*)$$

$$s_n^- \leq s_n \leq s_n^+ \quad \text{или} \quad \sum_{n=1}^N s_n^2 = S^2. \quad (16^*)$$

Связь между непрерывной задачей (14)–(16) и ее сеточной аппроксимацией (14*)–(16*) не нуждается в пояснениях; заметим лишь, что в дискретной задаче N есть произведение числа интервалов сетки на размерность управления. Задача (14*)–(16*) является либо задачей квадратического программирования, либо классической задачей на условный экстремум квадратичной формы в зависимости от того, какую форму имеют исходная задача и, соответственно, условие (16*).

Рассмотрим подробнее второй, классический вариант задачи, так как в этом случае метод множителей Лагранжа сводит задачу к системе линейных алгебраических уравнений. В самом деле, образовав функцию Лагранжа

$$L(s, \lambda) = \sum_{n=1}^N s_n h_n^0 + \frac{1}{2} \sum_{p=1}^N \sum_{q=1}^N H_{pq}^0 s_p s_q + \sum_{i=1}^m \lambda_i \sum_{n=1}^N s_n h_n^i + \lambda_0 \frac{1}{2} \sum_{n=1}^N s_n^2,$$

и приравняв нулю производные по s_n , получим систему линей-

ных алгебраических уравнений

$$\frac{\partial L}{\partial s_n} = \sum_{p=1}^N \frac{1}{2} (H_{pn}^0 + H_{np}^0) s_p + \lambda_0 s_n + \\ + \left\{ h_n^0 + \sum_{i=1}^m \lambda_i h_n^i \right\} = 0, \quad n = 1, 2, \dots, N. \quad (17)$$

Решая N уравнений (17) вместе с m уравнениями (15*) относительно $N + m$ неизвестных $\{s_n\}$, $\lambda_1, \lambda_2, \dots, \lambda_m$ при каком-то фиксированном значении λ_0 , получим решение, удовлетворяющее всем условиям задачи, за исключением (16*). Проведем подобные вычисления для нескольких значений λ_0 , подберем нужное значение λ_0 из условия (16*), которое, кстати, может быть удовлетворено с не очень высокой точностью. Однако самым неприятным моментом всего алгоритма является необходимость решения систем линейных алгебраических уравнений высокого порядка N . Этим объясняется, видимо, тот факт, что в известных автору работах метод второго порядка использовался на сравнительно грубых сетках с небольшим значением $N \approx 10 \div 20$. Если исходная вариационная задача содержит условие $u(t) \in U$, и в (16*) берется первый вариант ограничений на s_n , задача также оказывается вычислительно очень сложной при больших N . Таким образом, проявляется своеобразная противоречивость методов второго порядка. Имея целью в основном повысить эффективность поиска вблизи минимума и получить меньшее значение функционала, чем это удастся сделать методами первого порядка, методы второго порядка, реализованные на грубых сетках невысокой размерности, теряют в точности именно из-за грубости аппроксимации, из-за сужения задачи на пространство управлений, не допускающее очень точного приближения искомого оптимального $u(t)$.

Стоит упомянуть еще одну причину, по которой методы второго порядка представляются интересными. Это связано с выбором шага спуска S . В методах первого порядка эту величину приходится назначать, тогда как в методах второго порядка учет квадратичных членов разложения приводит к естественному выбору абсолютной величины вариации $\delta u(t)$ без введения искусственных ограничений.

Г Л А В А III

РЕШЕНИЕ ЗАДАЧ

§ 25. Общие замечания к третьей главе

Во второй главе были приведены характерные подходы к построению приближенных методов решения задач оптимального управления. Однако при их реализации возникает необходимость уточнить и конкретизировать ряд деталей. Кроме того, заранее не ясно, каковы будут затраты вычислительной работы, какие результаты удастся получить. Все это выясняется в процессе систематической эксплуатации метода. При решении конкретной задачи могут возникнуть какие-то специфические трудности, и нужно уметь разбираться в причинах возможных неудач, находить пути их преодоления. Совокупность подобных деталей образует низший уровень вычислительной технологии, однако, не владея им, не стоит браться за решение сложных задач. Перед автором стояла задача познакомить читателя и с этой стороной вопроса. Сейчас не видно другого способа сделать это, отличного от того, который реализован в настоящей главе. Здесь собраны примеры фактического решения прикладных задач оптимального управления, подробно показан и прокомментирован процесс их решения, разъяснены трудности, встретившиеся в той или иной из них, те конкретные приемы, которые были использованы, и достигаемый с их помощью эффект. Основу этой главы составляют задачи, решенные автором. Это естественно, так как по этим задачам автор располагает необходимым методическим материалом. Однако, если была возможность, автор привлекал и результаты расчетов, проведенных другими.

Нужно все-таки объяснить, почему в этой главе собрано сравнительно большое число примеров, и почему процесс решения задач так подробно освещается, зачем такое количество таблиц и графиков. Стремясь познакомить читателя с достаточно широким набором вычислительных приемов, методов анализа результатов и т. д., автор не хотел бы при их изложении ограничиться заверениями о том, что они оказались полезными, эффективными, позволили преодолеть какие-то трудности. Автор стремился показать, что трудности действительно были, и в чем они состояли, что их дей-

ствительно удалось преодолеть, и что это значит, какой смысл можно вкладывать в утверждения об «эффективном решении сложных задач». Все это так или иначе связано с характерной чертой вычислительной математики: в некотором смысле ее средства тривиальны и доступны не очень сведущему человеку. В то же время всякий, кто занимался вычислительной работой всерьез, знает, что она требует большого труда, фантазии и сообразительности, основательных теоретических знаний. Основной арсенал средств приближенных вычислений сформировался очень давно (например, метод Ньютона для решения уравнений, метод конечных разностей Эйлера и т. п.). Современное развитие этой науки можно условно разделить на две части: с одной стороны происходит обобщение основных идей на все более абстрактные ситуации и соответствующая модернизация классической терминологии. С другой стороны, попытки доведения классических идей до фактических расчетов обнаруживают их недостаточность, необходимость новых разработок, новых конструкций. Личные интересы автора связаны именно с этим вторым направлением развития вычислительных методов. Однако возникающие здесь вопросы не очень популярны и плохо освещены в теоретической литературе. Ведь они возникают лишь при доведении расчетов до ответа, удовлетворяющего заказчика. Читатель, наверное, заметит, что автор скептически относится к разработкам в области приближенных вычислений, не связанным с экспериментальной проверкой. Так же недоверчиво автор относится к утверждениям о создании «эффективного метода», смысл которых не разъяснен достаточно полными данными о решавшихся задачах, полученных результатах и затратах вычислительной работы. Ведь у автора таких утверждений и читателя могут быть совсем разные представления о том, что можно считать эффективным методом. В то же время упоминавшаяся уже видимая тривиальность вычислительных задач способствует появлению работ, авторы которых искренне уверены, что в вопросе приближенного решения задач оптимального управления, почему-то считающемся сложным, ими предложен метод, позволяющий справиться со всеми трудностями. Более того, иногда это подтверждается и ссылками на опыт вычислений. Вот характерный пример подобного рода, замечательный еще и тем, что приведены данные, позволяющие разобраться в том, что же в действительности удалось получить. В [75], [76] рассматривается задача оптимального управления: найти $\{u(t), x(t)\}$ из условий

$$\min_{u(\cdot)} \int_0^T \Phi^0(x, u) dt, \quad (1)$$

$\dot{x} = f(x, u), x(0) = X_0, x(T) = X_1, G[x(t), u(t)] \leq 0$ при всех t

(можно рассмотреть и более общую задачу). Оказывается, задачу можно очень просто решить, применяя методы математического программирования. На интервале $[0, T]$ вводится сетка $0 = t_0 < t_1 < \dots < t_N = T$ с переменным шагом $T_{i+1} = t_{i+1} - t_i$, функции заменяются своими сеточными проекциями $x_i = x(t_i)$, $u_i = u(t_i)$, уравнения — очевидными дискретными аппроксимациями. Получаем задачу:

$$\begin{aligned} \min \sum_{i=1}^N T_i \Phi(x_i, u_i), \\ x_{i+1} - x_i = T_{i+1} f(x_i, u_i), \quad x_0 = X_0, \quad x_N = X_1, \\ G(x_i, u_i) \leq 0, \quad i = 0, 1, \dots, N. \end{aligned} \quad (2)$$

Здесь искомыми считаются все величины x_i , u_i , T_i .

Задача (2) является хорошо изученной задачей математического программирования, для ее решения разработаны «эффективные методы», многие из которых оформлены в виде стандартных программ современного математического обеспечения ЭВМ. Остается только воспользоваться ими. Именно так и поступают авторы работ [75], [76] и получают решения нескольких задач; четыре из них представлены в [77] таблицами, позволяющими оценить результат. Разумеется, эти данные призваны убедить читателя в эффективности такого подхода. Если бы этим дело исчерпывалось, автору не следовало бы писать эту книгу, а утверждение о том, что занятие вычислительной математикой требует фантазии и теоретической подготовки, было бы явным преувеличением. В самом деле, составление уравнений (2) требует самых примитивных знаний, да и тот метод решения задачи (2), который был использован в [75] (мы еще вернемся к его обсуждению), тоже основан на не очень глубоких идеях. В конце концов важно знать, что такой метод есть, есть соответствующая стандартная программа, и нужно уметь ею воспользоваться. Обратимся, однако, к результатам. В [77] (стр. 211—214) рассматривается система с разностными уравнениями

$$\begin{aligned} x_{i+1}^1 - x_i^1 - T_{i+1} [-(x_i^1)^2 + x_i^2 + u_i] &= 0, \\ x_{i+1}^2 - x_i^2 - T_{i+1} x_i^1 &= 0. \end{aligned} \quad (3)$$

Аппроксимацию функционала, ограничений типа $x^1 \geq 0$, $x^2 \geq 0$ мы не выписываем: они очевидны и для дальнейшего несущественны. При $N=12$ задача (2) оказалась задачей минимизации с 48 переменными и 37 условиями. За восемь минут работы машины IBM-7094 (не уступающей по техническим данным нашей БЭСМ-6) был получен ответ. Он воспроизведен в табл. 1, заимствованной из [75] (см. также [77], стр. 213). В соответствии с (3) x_i^2 должна монотонно возрастать, в таблице же и на соответствующем ей графике x_i^2 монотонно падает. Несоответствие решения в таблице (1)

Т а б л и ц а 1

i	T_i	t_i	u_{i-1}	x_i^1	x_i^2
1	0,00291	0,00291	0,0553	0,00265	0,964
2	0,00196	0,00487	0,5000	0,00375	0,930
3	0,00176	0,00663	0,5000	0,00434	0,898
4	0,00169	0,00832	0,0466	0,00467	0,867
5	0,00164	0,00996	0,0435	0,00479	0,839
6	0,00162	0,01158	0,0419	0,00480	0,812
7	0,00159	0,01317	0,0406	0,00473	0,786
8	0,00156	0,01473	0,0395	0,00402	0,762
9	0,00151	0,01624	0,0383	0,00330	0,740
10	0,00144	0,01768	0,0370	0,00218	0,719
11	0,00130	0,01898	0,0350	0,00012	0,689
12	13,18224	13,20122	0,0339	9,26978	0,683

уравнению (3) настолько очевидно, что автор попытался объяснить его опечаткой и тем, что одноименные величины в уравнении и таблице отличаются масштабным множителем, о котором забыли сообщить. Для проверки этой гипотезы была вычислена величина $(x_{i+1}^2 - x_i^2)/(T_{i+1} x_i^1)$. Если бы она оказалась постоянной, гипотеза о забытом множителе была бы весьма правдоподобной. Однако эта величина меняется от -6500 при $i=1$ до -10 при $i=11$. Проверим, еще, например, первое из уравнений (3) для $i=4$: $0,00489 - 0,00467 - 0,00162 \times [-0,00467^2 + 0,867 + 0,0466] = 0,00022 - 0,00152 = -0,0013 \neq 0$. То же самое и в других точках i и во второй таблице, иллюстрирующей решение другого варианта задачи. В чем же дело? Попробуем все-таки объяснить это недоразумение. Разумеется, можно только высказать какое-то предположение, для большего опубликованные материалы не дают оснований. Видимо, дело в том методе, который использовался для решения задачи (2). Это — «метод последовательной минимизации без ограничений, разработанный Фиакко и Мак-Кормиком» ([77] стр. 214). Имеется в виду метод штрафных функций, в котором задача (2) сводится к задаче минимизации составного функционала J^* . Последний состоит из исходного минимизируемого функционала J , к которому добавлены с «коэффициентами штрафа» невязки в условиях задачи. Например,

$$J^*(x, u, T) \equiv J(x, u, T) + \sum_{i=1}^N K_i \|x_i - x_{i-1} - T_i f(x_i, u_i)\|^2 + \dots \quad (4)$$

В данной задаче $J \sim 100$, и для того, чтобы невязка (а в действительности, нужно добиваться невязки $\sim 0,00010 - 0,00001$) была хоть как-то заметна, нужно брать $K_i \simeq 10^7 - 10^{10}$. Видимо, K_i были заметно меньшими и невязки в уравнениях, будучи численно очень небольшими, содержательно очень велики, и ни о каком выполне-

нии соотношений (3) не может быть и речи. Сторонники данного подхода могут заметить, что такие ошибки еще не компрометируют самой идеи. Тем более, что результат легко, видимо, исправить: нужно снова решить задачу, умножив, например, все уравнения (3) на подходящий множитель или, что то же самое, увеличив коэффициенты штрафа. Но этот способ справиться с трудностями не так безобиден, как кажется. Ведь увеличение коэффициентов штрафа резко осложняет задачу минимизации J^* . Здесь автор хотел бы прекратить дискуссию с защитниками данного подхода. Ее имеет смысл продолжить лишь после того, как будут получены и предъявлены новые результаты. Ведь суть дела в количестве машинного времени, в качестве и надежности полученных результатов, а совсем не в принципиальной возможности решить задачу каким-то способом. Заметим лишь, что даже если и будут получены безупречные результаты для данной дискретной задачи, это еще не все, после этого можно будет перейти к следующему, более глубокому слою ошибок. Они обсуждаются в § 36 в связи с опубликованным в [75] и [77] (стр. 146—152) «решением» задачи управления отравлением ксеноном в ядерных реакторах. Таковы результаты упрощенного подхода к проблемам приближенного решения прикладных задач. А ведь «в принципе» этот подход вполне безупречен и обоснован теоретически: есть теоремы о том, что решение дискретной задачи (2) сколь угодно точно (при $N \rightarrow \infty$) аппроксимирует решение исходной задачи (1), есть (см. [61]) и теоремы о том, что задача $\min J^*$ сколь угодно точно (при $K \rightarrow \infty$) аппроксимирует задачу (2), есть, наконец, и теоремы о сходимости методов минимизации, с помощью которых находилась $\min J^*$. Однако остаются еще вопросы о том, можно ли данное N считать достаточно большим, можно ли используемые коэффициенты штрафа считать достаточно большими, можно ли ограничиться данным числом итераций в процессе минимизации J^* ? Другими словами, нужно уметь контролировать те результаты, которые выдает та или иная стандартная программа, особенно претендующая на решение такой задачи, как поиск минимума. Мы так подробно остановились на неудачных расчетах [77] потому, что они связаны с наметившейся в последнее время тенденцией трактовать вычислительную математику как умение пользоваться накопившимся обширным набором стандартных программ решения различных типов задач. В этом деле нужны большая осторожность и тщательный содержательный контроль результатов. Формальное описание этих стандартных программ обычно создает преувеличенное представление об их действительных возможностях, особенно у тех, кто не имеет достаточного вычислительного опыта. Используемый в [77] подход очень популярен, излагается во многих книгах и статьях, что создает видимость исчерпывающего решения задачи и свидетельствует о недостаточно ясном

понимании проблем в вычислительной математике. Это отнюдь не проблема принципиальной возможности приближенного решения (она в нашем случае тривиальна), а проблема фактической эффективности алгоритма. Вот почему автор так настороженно относится к публикациям, не содержащим подробного численного материала. Вот почему автор стремился предоставить читателю материал для самостоятельных суждений об эффективности метода, качестве решения и надежности результатов. Читатель должен также получить представление о трудоемкости расчетов. В связи с этим следует сделать разъяснение. В последнее время стало обычным приводить время решения задачи на ЭВМ. Автор этого не делает, предпочитая сообщать число итераций, достаточно подробные сведения о вычислениях, составляющих одну итерацию, и характер изменения основных величин в процессе итераций. Для этого есть веские причины. Почему не приводится машинное время — эта, казалось бы, основная характеристика эффективности? Дело в том, что время решения задачи очень сильно зависит от ряда привходящих обстоятельств. Например, от мощности ЭВМ (в настоящее время типов ЭВМ столько, что даже специалисты не всегда знают их технические характеристики). Далее, время зависит от шага интегрирования, от языка, на котором написана программа, от качества транслятора, и даже в рамках одного и того же транслятора часто можно сильно сократить время решения задачи, отказавшись от наиболее общих и удобных средств языка. Поэтому автор предпочитает указывать число итераций: именно эта характеристика наиболее полно отражает качество собственно метода *). Однако само по себе число итераций не дает еще полного представления об эффективности: ведь речь идет о процессе, в принципе, бесконечном, и важно знать, какие результаты на каком этапе получены. Эти замечания и определяют характер изложения материала в третьей главе. В этой главе автор неоднократно обращается и к расчетам, проделанным другими авторами и опубликованным как примеры удачного решения задач оптимального управления. В некоторых случаях более подробный анализ показывает, что расчеты были скорее неудачными. Стоит ли это делать? Нужна ли такая критика? Нужна, и вот почему. Читатель легко убедится, что приводимые автором расчеты (методом §§ 19—21) проходят, видимо, достаточно эффективно, но требуют привлечения не столь уж простых средств. В то же время в литературе предлагаются гораздо более простые алгоритмы, основанные на привычных большинству методах вычислений, и утверждается, что они дали хорошие и надежные результаты. Если это так, то, конечно же, следует предпочесть более

*) После такого разъяснения автор счел возможным привести и абсолютные цифры затрат машинного времени на решение некоторых задач.

простые методы. Автор, однако, неоднократно имел возможность убедиться, что заявления об удачном и надежном решении сложных задач относительно простыми средствами не соответствуют действительности. Иногда такие утверждения используют неопределенность и субъективность терминов «эффективный метод», «надежные результаты» и т. д. Однако может быть и другая причина, заслуживающая более внимательного анализа. Часто авторы подобных утверждений искренне уверены в их правоте. Ведь речь идет о решении задач с неизвестным ответом. Если принять меры предосторожности и гарантировать необходимую точность численного интегрирования системы $\dot{x}=f$ и вычисления функционалов задачи, можно утверждать, что найденное управление дает такое-то значение оптимизируемого функционала при выполнении всех условий задачи. Оно лучше некоторых управлений, взятых в качестве исходных приближений — за это тоже можно ручаться, а вот оптимально ли оно — это другой вопрос, тут определенных утверждений лучше не делать. Обычно результаты расчетов позволяют лишь утверждать, что продолжение поиска едва ли приведет к заметно лучшему результату. Но это может быть и следствием неэффективности метода. Критическое отношение к полученным результатам, сомнение в их правильности, искусство анализа и квалификации результата, как близкого к оптимальному, — это тоже необходимый элемент вычислительной работы, и автор хотел хотя бы в какой-то мере помочь читателю овладеть им. Этой цели и служит критический разбор некоторых решений. Конечно, читатель вправе спросить, почему автор считает свои результаты надежными и не подлежащими подобному же сомнению. Ведь в конце концов и он прекращает поиск по тем же соображениям, что и остальные: когда траектория практически уже не улучшается. Читатель убедится, что и свои решения автор старался проанализировать и получить более веские основания утверждать их практическую оптимальность, чем бесполезность продолжения поиска экстремума. Каковы же средства такого анализа? Их несколько, и все они так или иначе представлены в этой главе.

1. Проверка принципа максимума использовалась в задачах, в которых все функционалы дифференцируемы по Фреше.

2. Сравнение с точным (быть может, гипотетически точным) решением. Последнее бывает известно в тестовых задачах, и часто его удается угадать в результате анализа численных результатов.

3. Сравнение с результатами, полученными другими методами.

Разумеется, в наиболее сложных задачах доверие к результату основано в значительной степени на репутации метода. Если в многочисленных задачах, где решение так или иначе может быть проконтролировано, метод дал хорошие результаты, есть основа-

ния доверять решению и в более сложных ситуациях, не поддающихся подобному контролю. Этим тоже, в известной мере, объясняется большое число рассмотренных в этой главе задач. Приближенным решением задач оптимального управления занимались многие специалисты. Каждый из них имеет свою точку зрения на характер встречающихся в этой работе трудностей, свое объяснение тех или иных ситуаций в процессе решения задачи. Автор не всегда согласен с этими объяснениями и предлагает другие. Это ни в коем случае не проявление духа противоречия. Ведь правильная квалификация той или иной ситуации играет важную роль, от нее зависит, какие меры следует принять для преодоления трудностей. Например, если какая-то ситуация характеризуется как локальный минимум, ничего, в сущности, поделать нельзя. Однако если эта ситуация в действительности является тупиковой для используемого метода, выход следует искать в его совершенствовании.

§ 26. Задача о брахистохроне

Эта классическая задача послужит нам простым примером, позволяющим продемонстрировать некоторые аспекты численного решения вариационных задач. Мы будем рассматривать и решать ее в двух постановках — в классической и в современной.

Классическая постановка задачи. Найти функцию $x(t)$, $0 \leq t \leq T$, минимизирующую функционал

$$F[x(\cdot)] \equiv \int_0^T \sqrt{\frac{1 + \dot{x}^2}{x}} dt \equiv \int_0^T \Phi(x, \dot{x}) dt \quad (1)$$

и удовлетворяющую краевым условиям

$$x(0) = X_0, \quad x(T) = X_1. \quad (2)$$

Современная постановка задачи. Найти функцию $u(t)$, минимизирующую функционал

$$F_0[u(\cdot)] \equiv \int_0^T \sqrt{\frac{1 + u^2(t)}{x(t)}} dt, \quad (1^*)$$

определенный на решении системы

$$\dot{x} = u; \quad x(0) = X_0 \quad (2^*)$$

при дополнительном условии

$$F_1[u(\cdot)] \equiv x(T) - X_1 = 0. \quad (3^*)$$

Мы будем решать обе задачи методом спуска по градиенту. В том и другом случае речь идет о построении минимизирующей последовательности.

Метод градиента в классической задаче. Пусть $x^0(t)$ — некоторое исходное приближение, удовлетворяющее краевым условиям (2). Один шаг спуска состоит в следующем: строится направление спуска в функциональном пространстве — функция $y(t)$, удовлетворяющая краевым условиям $y(0)=0$, $y(T)=0$, и следующее, лучшее приближение ищется в виде

$$x^1(t) = x^0(t) + s^* y(t),$$

причем параметр s^* определяется одномерной задачей минимизации: определив функцию

$$R(s) \equiv F[x^0(\cdot) + sy(\cdot)], \quad (4)$$

найти

$$R(s^*) = \min_s R(s). \quad (5)$$

Фактическая реализация этой схемы требует решения еще двух вопросов.

Конечномерная аппроксимация задачи. Тут возможны различные способы; мы остановимся на методе сеток. Интервал $[0, T]$ покрывается счетной сеткой точек с равным, для простоты, шагом τ :

$$t_0 = 0 < t_1 < t_2 < \dots < t_N = T; \quad t_n = n\tau, \quad \tau = T/N.$$

Искомую функцию будем задавать значениями в узлах $x_n \sim x(t_n)$, между узлами сетки будем считать $x(t)$ линейной. Функционал аппроксимируем интегральной суммой *)

$$F[x] = \tau \sum_{n=0}^{N-1} \sqrt{1 + \left(\frac{x_{n+1} - x_n}{\tau}\right)^2} \left| \frac{(x_n + x_{n+1})}{2} \right|. \quad (6)$$

Заметим сразу же, что эта аппроксимация хотя и выглядит совершенно естественной, таит в себе возможности грубых ошибок; ниже мы обсудим причины этого и внесем необходимые изменения.

Выбор направления спуска $y(t)$. Естественным представляется спуск по направлению градиента функционала. Градиент $F[x(\cdot)]$ вычисляется с помощью оператора Эйлера

$$\begin{aligned} y(t) &= -\frac{\partial F[x(\cdot)]}{\partial x(\cdot)} = \frac{d}{dt} \Phi_{\dot{x}}[x(t), \dot{x}(t)] - \Phi_x[x(t), \dot{x}(t)] = \\ &= \frac{\dot{x}(t) + [1 + \dot{x}^2(t)]/2x(t)}{\sqrt{[1 + \dot{x}^2(t)]^3} x(t)}. \end{aligned} \quad (7)$$

*) Совокупность значений $\{x_n\}_{n=0}^N$ будем обозначать x_\cdot .

Теперь возникают две возможности: либо формально аппроксимировать выражение (7) на введенной сетке, например, определяя сеточную функцию y_n формулой (7) и полагая в точке $n=1, 2, \dots, N-1$:

$$\ddot{x}_n = \frac{1}{\tau^2} (x_{n+1} - 2x_n + x_{n-1}), \quad \dot{x}_n = \frac{1}{2\tau} (x_{n+1} - x_{n-1}), \quad (7^*)$$

$$y_0 = 0; \quad y_N = 0,$$

либо, предвидя некоторые трудности и стремясь избежать их, выбрать специальную аппроксимацию (7), согласованную с аппроксимацией функционала (6) в следующем смысле:

$$y_n = - \frac{\partial F(x_n)}{\partial x_n}, \quad n = 1, 2, \dots, N-1, \quad y_0 = y_N = 0. \quad (7^{**})$$

Расчетные формулы для y_n имеют вид

$$\begin{aligned} a_{n+1/2} &= x_{n+1} - x_n; \quad b_{n+1/2} = x_{n+1} + x_n, \\ y_n &= \frac{a_{n-1/2}}{\sqrt{b_{n-1/2}(\tau^2 + a_{n-1/2}^2)}} - \frac{1}{2b_{n-1/2}} \sqrt{\frac{\tau^2 + a_{n-1/2}^2}{b_{n-1/2}}} - \\ &\quad - \frac{a_{n+1/2}}{\sqrt{b_{n+1/2}(\tau^2 + a_{n+1/2}^2)}} + \frac{1}{2b_{n+1/2}} \sqrt{\frac{\tau^2 + a_{n+1/2}^2}{b_{n+1/2}}}. \end{aligned}$$

Что же получается при фактической реализации этой расчетной схемы? Для иллюстрации была выбрана конкретная задача: $T=2$, $x(0)=3$; $x(2)=10$, $N=100$. (В этом случае аппроксимация (6) удовлетворительна; неприятности были бы в задаче с $x(0)=0$; ее мы тоже рассмотрим.) Решение задачи может быть найдено:

$$\min_{x(\cdot)} F[x(\cdot)] = 2,9715 \dots$$

Расчет I. В качестве начального приближения выбрана функция

$$x^0(t) = \begin{cases} 3 & \text{при } 0 \leq t \leq 1, \\ 3 + 7(t-1) & \text{при } 1 \leq t \leq 2. \end{cases}$$

Направление спуска $\{y_n\}$ строилось с помощью формальной аппроксимации (7*). Результаты эксперимента представлены в табл. 1, где указаны

- 1) ν — номер приближения;
- 2) $F[x^\nu(\cdot)]$ — значение функционала на $x^\nu(\cdot)$;
- 3) n — число вычислений функционала $F[x^\nu(\cdot) + sy(\cdot)]$ при разных значениях s , потребовавшееся для определения «оптимального» шага s^* решением задачи типа (5).

Видно, что процесс строит минимизирующую последовательность сеточных функций, однако значения $F[x^\nu]$ убывают столь

Т а б л и ц а 1

I			III			IV		
ν	F_0	n	ν	F_0	n	ν	F_0	n
0	3,46653	5	0	3,00466	4	0	3,00466	8
1	3,45733	7	1	3,00430	5	1	2,99202	5
2	3,45215	5	2	3,00408	5	2	2,98621	5
3	3,44954	5	3	3,00401	4	3	2,98593	5
4	3,44815	5	4	3,00397	4	4	2,98579	5
5	3,44705	5	5	3,00394	5	5	2,98576	5
6	3,44609	5	6	3,00391	5	6	2,98573	5
7	3,44520	5	7	3,00388	5	7	2,98571	5
8	3,44437	5	8	3,00386	5	8	2,98568	5
9	3,44358	5	9	3,00384	5	9	2,98566	5
10	3,44282		10	3,00382		10	2,98564	

V			VI			VII			VIII		
ν	F_0	n	ν	F_0	n	ν	F_0	n	ν	F_0	n
0	3,46653	9	0	3,00466	8	0	3,46653	8	0	3,00466	7
1	2,99109	4	1	2,97447	4	1	2,99651	6	1	2,97157	7
2	2,98916	7	2	2,97152	4	2	2,97156	4	2	2,97152	7
3	2,97652	5	3	2,97141	5	3	2,97134	4	3	2,97152	
4	2,97140	5	4	2,97138	4	4	2,97133				
5	2,97134	4	5	3,97138							
6	2,97133										

медленно, что достижение приемлемой точности представляется довольно сомнительным (при реальных затратах машинного времени). Попробуем разобраться в возможных причинах и исправить положение.

Первое предположение: быть может, следует использовать согласованную аппроксимацию (7**)?

Р а с ч е т II. Он отличается от I расчета только тем, что u_n вычислялись по формулам (7**.*). Результаты показывают, что положение не изменилось.

Второе предположение: в § 1 уже отмечалось, что следует ожидать определенных неприятностей в связи с неограниченностью оператора Эйлера. В самом деле, на рассматриваемом начальном приближении градиент $\partial F[x(\cdot)]/\partial x(\cdot)$ содержит δ -функцию с полюсом в точке $t=1$. Попытка разобраться в точной постановке задачи, вычисляя $F[x^0(\cdot) - sy(\cdot)]$, приводит к серьезным

*) Этот расчет в таблице не представлен. Ничего не дает и увеличение точности определения s из (5).

затруднениям: нужно возвести производную δ -функции в квадрат, разделить это на δ -функцию, из результата извлечь квадратный корень. В разностной аппроксимации дело несколько проще: δ -функция имеет в полюсе значение $\sim s/\tau$; ее производная $\sim s/\tau^2$, квадрат последней s^2/τ^4 . Итак, при вычислении интегральной суммы (6) будем иметь, по крайней мере, одно слагаемое порядка

$$\tau \sqrt{\frac{s^2}{\tau^4} \frac{s}{\tau}} = \sqrt{\frac{s}{\tau}}.$$

Ясно, что расчет может протекать лишь при очень малых шагах $s \ll \tau$, что и видно по результатам.

Третье предположение: трудности связаны с заведомо неудачным выбором начального приближения. Попробуем выбрать другое, более разумное и естественное (заметим, что в данной задаче это сделать можно, так как качественный характер искомого решения нам известен).

Р а с ч е т III. От I расчета он отличается лишь выбором начального приближения

$$x^0(t) = 3 + 3, 5t, \quad 0 \leq t \leq 2.$$

Представленные в таблице результаты показывают, что положение несколько улучшилось, но сходимость по-прежнему слишком медленна.

Четвертое предположение: от функции $y(t)$ мы требуем краевых условий $y(0) = y(T) = 0$; однако на данном приближении $x^0(t)$, как нетрудно убедиться, функция $y(t) = -\frac{\partial F[x(\cdot)]}{\partial x(\cdot)}$ при $t=0$, $t=T$ в 0 не обращается, что в разностной реализации (7) или (7*) приводит к разрывам порядка s в функции $x(t) + sy(t)$ на краях интервала; разрыв при дифференцировании дает величину $\sim s/\tau$ и т. д. с примерно теми же неприятными последствиями, что и в первом случае. Исправить положение легко; после вычисления y_n по формулам (7) или (7*) пересчитаем их:

$$y_n = y_n t_n (2 - t_n) \quad (8)$$

(умножение градиента на положительную функцию допустимо).

Р а с ч е т IV. Отличаясь от III расчета только преобразованием (8), он показывает, что сходимость стала заметно лучше; однако особого оптимизма результаты не внушают, особенно если иметь в виду использование метода в более сложных ситуациях, где предварительная информация о характере решения не позволяет выбрать столь же хорошее начальное приближение, как в данном случае *).

*) Столь медленная сходимость, как в расчетах I—IV, существенно связана с числом N , так как норма разностной аппроксимации оператора $\partial F/\partial x(\cdot)$ есть $O(N^2)$. На более грубой сетке сходимость более быстрая.

Пятое предположение: плохая сходимость метода спуска по градиенту связана с тем, что градиент вычисляется применением к исходному приближению $x^0(t)$ неограниченного оператора Эйлера, эквивалентного, грубо говоря, оператору d^2/dt^2 ; умножение его на положительную функцию типа (8) приводит к такому же, в сущности, оператору $t(T-t)d^2/dt^2$ и не решает проблемы. Попробуем умножить градиент на положительный оператор, который в основной своей дифференциальной части был бы «обратен» оператору Эйлера. Так приходим к направлению спуска

$$y(t) = \left(-\frac{d^2}{dt^2}\right)^{-1} \left[\frac{d}{dt} \Phi_x - \Phi_x\right]. \quad (9)$$

Конкретная реализация этой идеи состоит в следующем: после вычисления градиента (7*) или (7**) направление спуска \tilde{y}_n получаем решением простой краевой задачи

$$\frac{\tilde{y}_{n+1} - 2\tilde{y}_n + \tilde{y}_{n-1}}{\tau^2} = y_n; \quad n = 1, 2, \dots, N-1, \quad y_0 = y_N = 0. \quad (9^*)$$

Р а с ч е т V. Он отличается от I расчета использованием формулы (9).

Этим же отличается от III расчет VI.

Результаты достаточно красноречивы и нет необходимости их комментировать; заметим лишь, что в этом случае совершенно несущественно, какая аппроксимация выбрана для градиента Эйлера — формальная (7*) или (7**), согласованная с (6).

Р е ш е н и е з а д а ч и о б р а х и с т о х р о н е в с о в р е м е н н о й п о с т а н о в к е. Пусть $u^0(\cdot)$ — исходное приближение, допустимое в том смысле, что решение задачи $\dot{x} = u$, $x(0) = X_0$ удовлетворяет второму краевому условию $x(T) = X_1$. Вычислим производные функционалов F_0 и F_1 :

$$\begin{aligned} \delta F_0[\delta u(\cdot)] &= \int_0^T w_0(t) \delta u(t) dt; \\ \delta F_1[\delta u(\cdot)] &= \int_0^T w_1(t) \delta u(t) dt, \end{aligned}$$

где

$$w_0(t) = \psi(t) + \frac{u(t)}{\sqrt{x(t)[1+u^2(t)]}}, \quad w_1(t) \equiv 1,$$

а $\psi(t)$ — решение задачи

$$\frac{d\psi}{dt} = \frac{1}{2} \frac{\sqrt{1+u^2(t)}}{x^{3/2}(t)}; \quad \psi(T) = 0.$$

В данном случае направлением спуска $\delta u(t)$ будет проекция градиента $w_0(t)$ на «гиперплоскость» в функциональном пространстве, определяемую условием]

$$\int_0^T \delta u(t) dt = 0.$$

Проще говоря, речь идет об условном градиенте, вычисление которого осуществляется так: положим (следуя правилу множителей Лагранжа; см. § 45)

$$\delta u(t) = w_0(t) - \lambda w_1(t).$$

Величина λ находится из условия

$$\int_0^T w_1(t) \delta u(t) dt = \int_0^T [w_0 w_1 - \lambda w_1 w_1] dt = 0,$$

т. е.

$$\lambda = \frac{\int_0^T w_0(t) w_1(t) dt}{\int_0^T w_1(t) w_1(t) dt}.$$

Далее вводим однопараметрическое семейство управлений $u(\cdot)$ — $s \delta u(\cdot)$, и параметр s находим решением задачи

$$\min_s R(s), \quad R(s) \equiv F_0[u(\cdot) - s \delta u(\cdot)].$$

Расчет VII. Он соответствует расчету I в смысле начальных данных

$$u(t) = \begin{cases} 0 & \text{при } 0 \leq t \leq 1, \\ 7 & \text{при } 1 < t < 2. \end{cases}$$

Эффективность процесса построения минимизирующей последовательности так же высока, как и в расчете V*).

Обратим внимание на то, что в процессе вычисления функциональных производных $w_0(t)$, $w_1(t)$ нам нигде не пришлось использовать операции дифференцирования по t компонент исходной невозмущенной траектории $\{u(\cdot), x(\cdot)\}$.

*) Расчеты V—VIII контролировались проверкой необходимого условия оптимальности: в конце этих расчетов норма градиента была в $\approx 10^6$ раз меньше, чем в начале. В расчетах I—IV она почти не менялась. Таким образом, найденные траектории с хорошей точностью удовлетворяют уравнению Эйлера (принципу максимума).

По этой же схеме решим задачу с данными

$$T=2; \quad X_0=0; \quad X_1=1, 2, \quad \min F_0[x(\cdot)] = 3,547 \dots$$

Ее отличие от первого варианта связано со значением $X_0=0$, что приводит к существенной погрешности аппроксимации (6). Дело в том, что точность (6) предполагает наличие у функции непрерывной первой производной. В случае задачи с $X_0 > 0$ искомое решение, как известно, обладает необходимым запасом гладкости. Однако при $X_0=0$ оно имеет особенность при $t=0$ типа \sqrt{t} , что дает в производной бесконечность типа $t^{-1/2}$. Это приводит к полному искажению численного решения. В самом деле, рассмотрим сеточную функцию следующего вида:

$$x_0=0, \quad x_1=x_2=x_3=\dots=x_N=X_1=1,2. \quad (10)$$

Значение функционала (6) легко подсчитывается:

$$F[x_0] = \tau \sqrt{\frac{1 + \left(\frac{1,2}{\tau}\right)^2}{\frac{1}{2} \cdot 1,2}} + (N-1) \tau \sqrt{\frac{1}{1,2}} \approx 3,35,$$

что заметно меньше точного минимального значения $F=3,547$. Расчеты, использующие формулу (6), в этом случае приводили к немедленному сползанию численного решения к функции (10).

Легко исправить этот дефект. Фактическая реализация метода осуществлялась на той же временной сетке, причем в качестве управлений $u(t)$ брались кусочно постоянные функции

$$u(t) = u_{n+1/2} \quad \text{при} \quad t_n \leq t < t_{n+1}, \quad n = 0, 1, \dots, N-1. \quad (11)$$

Тогда решение $\dot{x}=u$ есть кусочно линейная функция, и функционал вычисляется точным интегрированием в этом классе функций:

$$F[u(\cdot)] = \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} \sqrt{\frac{1 + u_{n+1/2}^2}{x_n + (t - t_n) u_{n+1/2}}} dt, \quad (12)$$

что после несложных преобразований приводит к расчетной формуле:

$$F[u(\cdot)] = \sum_{n=0}^{N-1} \begin{cases} \tau \sqrt{\frac{1 + u_{n+1/2}^2}{x_n + x_{n+1}}}, & \text{если} \quad |u_{n+1/2}| < \varepsilon \sim 0, \\ \frac{2\sqrt{1 + u_{n+1/2}^2}}{u_{n+1/2}} (\sqrt{x_{n+1}} - \sqrt{x_n}) & \text{при} \quad |u_{n+1/2}| \geq \varepsilon. \end{cases} \quad (13)$$

Процесс численного решения этой задачи (на сетке с $N=50$) показан в табл. 2, где представлены четыре шага спуска по условному градиенту. Таблица показывает, как осуществлялся подбор

Т а б л и ц а 2

1-й спуск	s	0	1,25	2,50	3,51	3,65				
	R	4,26	3,89	3,74	3,673	3,666				
2-й спуск	s	0	1,2	2,4	4,16	3,32	4,01	18,0	20,3	22,1
	R	3,666	3,653	3,641	3,626	3,633	3,627	3,559	3,556	3,555
3-й спуск	s	0	7,3	14,7	4,3	5,9	5,2	-6,7	312	78
	R	3,555	3,5547	3,5543	3,5549	3,5548	3,5548	3,5556	3,575	3,5529
4-й спуск	s	0	25	50	15,2		14,2			
	R	3,5529	3,55256	3,55259	3,55266		3,55268			

шага спуска s алгоритмом параболической аппроксимации (см. § 45). Исходное приближение имело вид

$$u^0(t) \equiv X_1/T.$$

Видно, что уже второй шаг спуска дает практически окончательный ответ; на третьем и четвертом шагах алгоритм поиска s^* работает не очень уверенно; это связано с потерей точности из-за сокращения значащих цифр при вычислении второй разностной производной (расчет проводился на машине с 29-значной двоичной мантиссой и с машинным нулем $\sim 10^{-9}$, все предыдущие — на БЭСМ-6, с 40-значной мантиссой и нулем $\sim 10^{-20}$).

Обсудим вопрос о различии точного значения $F_0 = 3,547$ и полученного численно $F_0 = 3,553$.

В этой задаче (и это характерно для всех вообще вариационных задач) ошибка численного решения естественно распадается на две части.

1. *Ошибка аппроксимации*, источником которой является замена дифференциальной постановки задачи аппроксимирующей конечно-разностной. Эта ошибка легко может быть уменьшена за счет, например, измельчения шага сетки и, в какой-то мере, за счет использования более точных разностных уравнений. Последняя оговорка не случайна: используемая нами аппроксимация функционала F_0 (13) точна в классе кусочно постоянных $u(t)$ и, если оставаться в этом классе, дальнейшее повышение точности аппроксимации невозможно без увеличения числа N интервалов постоянства u ; если же мы попытаемся использовать в расчетах другой класс функций $u(t)$, дающий более высокую точность аппроксимации гладких функций (а нам известно, что искомое

решение — достаточно гладкое, если не считать особенности (\sqrt{t}) , то придется заметно пересмотреть и процедуру численного решения; она, естественно, усложняется.

2. *Ошибка поиска*, источником которой является то, что процесс построения минимизирующей последовательности не доводится до конца; уменьшение этой ошибки в известной мере — вопрос машинного времени. Однако и здесь не случайно появилась оговорка: только за счет продолжения расчета ошибку поиска нельзя сделать сколь угодно малой: ведь поиск использует градиент, последний вычисляется приближенно; по мере приближения численного решения к минимуму градиент стремится к нулю, входящие в него конечные слагаемые взаимно уничтожаются, происходит сокращение главных знаков и в остатке, который, собственно, и идет в вычисления, все большую роль начинают играть всевозможные ошибки приближенных методов. Поэтому, не повышая точности промежуточных вычислений, нельзя сделать ошибку поиска сколь угодно малой.

Однако можно с достаточными основаниями утверждать, что обычно в прикладных расчетах ошибка аппроксимации легче поддается контролю вычислителя и не доставляет ему особых хлопот: она, как правило, много меньше ошибки поиска.

В данной задаче можно оценить ошибку аппроксимации (она в основном образуется на первом счетном интервале, в районе особенности решения). Подсчет показал, что в разность

$$\min F_0^{\text{числ}} - \min F_0^{\text{точн}} = 0,006$$

вклад ошибки аппроксимации (при $N=50$) есть 0,005, а остальная часть $\sim 0,001$ — вклад ошибки поиска.

При решении задачи обнаружилось досадное обстоятельство: аппроксимация (6), казавшаяся совершенно естественной, привела к численному результату, не имеющему содержательного смысла. В данной задаче, поскольку ее решение хорошо известно, можно было предвидеть это и легко указать способ исправления формул разностной аппроксимации. В связи с этим возникают два вопроса: как можно было бы в более сложной ситуации, когда решение нам неизвестно даже качественно, получив численное решение (10), догадаться о его ошибочности, и какие меры могли бы предупредить его появление? Ответ на первый вопрос не совсем прост. Конечно, если бы мы располагали теоремой о непрерывности искомого решения $x(t)$, сеточная функция (10), разумеется, вызвала бы подозрение о какой-то вычислительной ошибке. Однако в принципе появление разрывов в фазовой траектории $x(t)$ (и, следовательно, особенности типа δ -функции в управлении $u(t)$) возможно: мы встретимся с подобным явлением не только в искусственной модельной задаче § 35, но и в имеющих содержательный смысл задачах §§ 29, 34. Поэтому решение типа (10) должно

отвергаться или приниматься после соответствующей проверки. Достаточно общий способ подобной проверки состоит в следующем: получив соответствующее (10) управление, являющееся кусочно постоянной функцией на сетке с шагом Δt , следует проинтегрировать систему $\dot{x}=f(x, u)$ и вычислить функционалы задачи с шагом, много меньшим Δt . Это прояснило бы ситуацию, так как для $x(t)$ вида (10) было бы получено значение F_0 не 3,35, а, как по разностной формуле (13), $F_0 \simeq 4$. В классической постановке задачи мы получили бы то же самое, если бы, восполнив (10) до непрерывной функции линейной интерполяцией, вычислили значение интеграла по той же самой формуле (6), но с шагом, много меньшим шага сетки Δt . Таким образом, если бы мы придерживались техники решения задач, в которой шаг интегрирования системы $\dot{x}=f(x, u)$ обычно много меньше шага сетки для u , решение вида (10) не появилось бы. Заметим еще, что при решении первой задачи, используя аппроксимацию (6), мы получили численное значение $\min F_0=2,97133$, что несколько меньше точного значения $\min F_0=2,9715$. Это есть следствие ошибки аппроксимации. Во второй задаче использовалось вычисление F_0 точным интегрированием в классе кусочно постоянных $u(t)$. Поэтому численное значение $\min F_0$ может быть только больше точного.

Решение задачи методом сопряженных градиентов описано в [62] (оттуда и заимствованы значения $T=2$, $X_0=0$, $X_1=1, 2$). Использовалась аппроксимация типа (6) (при $N=50$) и задача решалась, как конечномерная задача поиска минимума. За 370 итераций метода сопряженных градиентов (1508 вычислений градиента) получено решение с 9-ю знаками F и 8-ю знаками x .

Но гораздо интереснее было бы знать, сколько итераций дает решение с 3—4 знаками. Именно от этого зависит оценка метода как средства решения прикладных задач. Получение же девяти знаков есть результат скорее спортивного, чем прикладного значения. К сожалению, этих данных в [62] нет. Неправильная аппроксимация (6) также снижает методическую ценность этих расчетов.

§ 27. Линейная задача быстрогодействия

Эта простая задача использовалась рядом авторов в качестве теста, на котором отрабатывались предлагаемые ими методы приближенного решения. Итак, рассматривается управляемая система

$$\dot{x}^1 = x^2; \quad \dot{x}^2 = x^3; \quad \dot{x}^3 = u; \quad x(0) = 0, \quad (1)$$

условие $u \in U$ имеет вид $|u| \leq 1$.

Задача состоит в определении управления $u(t)$, минимизирующего время T перевода точки x из $x(0)=0$ в заданную точку X :

$x(T)=X$. В работе А. Я. Дубовицкого и В. А. Рубцова [30] эта задача сводилась к краевой задаче для П-системы, решением которой является оптимальная траектория.

Для преодоления некоторых вычислительных трудностей в этих расчетах использовалась строго выпуклая аппроксимация, в результате которой П-система приобрела форму:

$$\begin{aligned} 1) \quad \dot{x}^1 &= T(x^2 + u_1), \quad \dot{x}^2 = T(x^3 + u_2), \quad \dot{x}^3 = Tu_3, \quad x(0) = 0; \\ 2) \quad \dot{\psi}_1 &= 0; \quad \dot{\psi}_2 = -T\psi_1; \quad \dot{\psi}_3 = -T\psi_2, \quad 0 \leq t \leq 1; \end{aligned} \quad (2)$$

$$3) \text{ уравнение (принцип максимума) для } u(t) - \\ (u(t), \psi(t)) = \max_{u \in U} (u, \psi(t)). \quad (3)$$

Область U в расчетах бралась как строго выпуклая аппроксимация первоначальной:

$$u \in U: \quad u_3^2 + A^2(u_1^2 + u_2^2) \leq 1, \quad A \gg 1. \quad (4)$$

Уравнение для $u(t)$ — однозначно разрешимо, и алгоритм решения краевой задачи для П-системы был реализован следующим образом.

Искомый набором «управляющих» параметров были величины $\alpha = \{\alpha_1, \alpha_2, \alpha_3\}$, $\alpha_1 = T$, $\alpha_2 = \psi_2(0)$, $\alpha_3 = \psi_3(0)$; значение $\psi_1(0) = 1$ фиксировано, так как вектор ψ определен с точностью до положительного множителя. Задание α замыкает уравнения 1), 2) до задачи Коши, которая решалась методом конечных разностей по следующей схеме перехода от $t_n = n\tau$ к $t_{n+1} = t_n + \tau$ ($\tau = 1/N$; N — число шагов):

$$\begin{aligned} \text{а) определяется } u(t_n) \text{ решением уравнения} \\ (u(t_n), \psi(t_n)) = \max_{u \in U} (u, \psi(t_n)), \end{aligned} \quad (5)$$

б) $x(t_{n+1})$, $\psi(t_{n+1})$ находятся из системы разностных уравнений следующего вида:

$$\frac{x^1(t_{n+1}) - x^1(t_n)}{\tau} = T \frac{x^2(t_{n+1}) + x^2(t_n)}{2} + Tu(t_n), \quad (6)$$

остальные уравнения аппроксимируются по той же схеме.

Правый конец траектории $x(t_N)$ становится, таким образом, однозначной функцией вектора α : $x(t_N) = Z(\alpha)$, и затем дело сводится к решению системы трех уравнений: $Z(\alpha) = X$.

Авторы этих расчетов отмечают следующие обстоятельства.

1. Для уравнений 1) и 2) использовались так называемые согласованные аппроксимации (см. § 5).

2. Строго выпуклая аппроксимация приводит к сглаживанию разрывов в функции $u_3(t)$ (только она и имеет содержательный смысл); это сглаживание зависит, разумеется, от величины A и определенным образом влияет на величину шага интегрирования

Грубо говоря, нужно, чтобы разрыв в $u_3(t)$ был «размазан» на достаточное число счетных точек; если это число было слишком мало, появлялись трудности при решении системы нелинейных уравнений $Z(\alpha) = X$: сходимость итерационного процесса решения этих уравнений становилась ненадежной, медленной. Причины этого мы обсудим ниже, при описании проводившихся автором экспериментов по этой же задаче.

Расчеты автора проводились по той же схеме, что и расчеты Дубовицкого и Рубцова, отличаясь лишь в следующих технических деталях.

1. Задача решалась в первоначальной постановке (1), поэтому решение уравнения (3) имело простой вид: $u(t) = \text{sign } \psi_3(t)$.

2. Численное интегрирование П-системы осуществлялось следующим образом: сначала интегрировались уравнения для $\phi(t)$ конечноразностным методом с шагом $\tau = 0,01$; вид разностных уравнений совершенно несуществен. Затем определялась функция $u(t)$ по значениям $\psi_3(t_n)$ на сетке $\{t_0, t_1, \dots, t_N\}$.

В этом месте появляется основное отличие от расчетной схемы (6): функция $u(t)$ на интервале $[t_n, t_{n+1}]$ полагалась равной $\text{sign } \psi_3(t_n)$, если $\text{sign } \psi_3(t_n) = \text{sign } \psi_3(t_{n+1})$; если же $\psi_3(t)$ меняет знак между точками t_n и t_{n+1} , находился корень уравнения $\psi_3(t_n + \xi\tau) = 0$, число $0 \leq \xi \leq 1$ находилось по линейной интерполяции $\psi_3(t)$ с узлов t_n, t_{n+1} ; теперь на (t_n, t_{n+1}) полагалось

$$u(t) = \xi \text{sign } \psi_3(t_n) + (1 - \xi) \text{sign } \psi_3(t_{n+1}). \quad (7)$$

Поясним значение этого усовершенствования для успеха всего расчета. Дело в том, что значения вектора α влияют на значения $x(T)$ не прямо, а через положение нулей функции $\psi_3(t)$. В разностной схеме типа (5)–(6) зависимость $Z(\alpha)$ носит «ступенчатый» характер: пока изменения параметров α_2 и α_3 малы настолько, что нули $\psi_3(t)$ перемещаются в пределах одного счетного интервала сетки, эти изменения не влияют на $x(T)$ — влияние проявляется скачком при переходе нуля $\psi_3(t)$ через узел сетки. Разумеется, эти скачки имеют размеры $\sim \tau$, однако, например, метод Ньютона основан на линеаризации зависимости $Z(\alpha)$ в окрестности некоторой точки: $Z(\alpha + \delta\alpha) \approx Z(\alpha) + Z_\alpha \delta\alpha$, а при ступенчатой зависимости Z от α в этой формуле появляются значительные погрешности, что и приводит к вычислительным трудностям. В схеме (5)–(6) сглаживание разрывов приводит к сглаживанию зависимости $Z(\alpha)$; в наших расчетах это достигается использованием формулы (7).

Что касается используемой в методе Ньютона матрицы Z_α , то она находилась численным дифференцированием по односторонней разностной формуле. Процесс решения уравнений $Z(\alpha) - X = 0$ проводился по схеме модифицированного метода Ньютона: в некоторой точке α^0 определялись $\delta\alpha = -Z_\alpha^{-1}(\alpha^0) [Z - X]$

и следующее приближение $\alpha^1 = \alpha^0 + s^* \delta \alpha$, где s^* — точка минимума $\|Z(\alpha^0 + s \delta \alpha) - X\|$. Рассмотрим процесс фактического решения по этой схеме задачи при $X = \{16, 0, 0\}$. В качестве исходного приближения был взят вектор $\alpha^0 = \{4,33; 1,1; 0,5\}$. Система уравнений $Z_\alpha(\alpha^0) \delta \alpha = -[Z - X]$ имеет вид

$$\begin{pmatrix} 3,64 & -41,6 & 50,5 \\ 2,58 & -27,1 & 30,7 \\ 1,01 & -9,05 & 9,52 \end{pmatrix} \begin{Bmatrix} \delta \alpha_1 \\ \delta \alpha_2 \\ \delta \alpha_3 \end{Bmatrix} = \begin{Bmatrix} 12,09 \\ -3,46 \\ -2,50 \end{Bmatrix}. \quad (8)$$

Ее решение ($\delta \alpha_1 = \delta T = 30,0$; $\delta \alpha_2 = \delta \psi_2(0) = 11,9$; $\delta \alpha_3 = \delta \psi_3(0) \approx 1$) определяет направление спуска $\|Z(\alpha)\|$. Оказалось, что это направление — в силу вычислительных ошибок при определении Z_α численным дифференцированием — крайне неэффективно при непосредственном определении нормы

$$\|Z(\alpha) - X\|_\mu^2 = \sum_{i=1}^3 [x^i(T) - X^i]^2.$$

Уже при беглом взгляде на систему (8) видна неравноправность компонент $Z(\alpha)$: при одном и том же смещении $\delta \alpha$ справедливо неравенство $|\delta x^1(T)| \gg |\delta x^3(T)|$.

В методе Ньютона направление спуска $\delta \alpha$ определяется так, что вдоль него убывают все компоненты «невязки» $Z(\alpha + \delta \alpha) - X$, однако это убывание в силу нелинейности задачи постепенно замедляется и сменяется ростом; в настоящей задаче именно так обстоит дело с компонентой $x^1(T) - X^1$, причем ее падение при очень малых s сменяется ростом, а в силу большого веса (см. (8)) этой компоненты начинается рост $\|Z(\alpha^0 + s \delta \alpha) - X\|$ при очень малых s . Эта ситуация усугубляется еще и влиянием ошибок численного дифференцирования при нахождении матрицы Z_α , так что найденное первое направление спуска оказалось даже направлением слабого роста $\|Z(\alpha) - X\|$. Положение в корне изменилось после введения масштабов в определение нормы так, как это описано в § 19:

$$\|Z\|_\mu^2 = \mu_1^2 (Z^1)^2 + \mu_2^2 (Z^2)^2 + \mu_3^2 (Z^3)^2.$$

На первом шаге процесса множители (они, разумеется, зависят от точки α) были следующими:

$$\mu_1 : \mu_2 : \mu_3 = 0,23 : 0,59 : 5,9.$$

Таким же примерно оставалось их отношение до конца поиска, процесс которого отражен в табл. 1 следующими величинами: номер итерации ν и полученные на этом этапе процесса поиска $\psi_2(0)$, $\psi_3(0)$, T , $x^1(T)$, $x^2(T)$, $x^3(T)$, $F = \|Z - X\|_\mu$, t_1 и t_2 — нули функции $\psi_3(t)$ (моменты переключения управления), n — число интегрирований задачи Коши для П-системы, понадобившееся

Таблица 1

ν	$\psi_2(0)$	$\psi_3(0)$	T	$x^1(T)$	$x^2(T)$	$x^3(T)$	F	n	t_1	t_2	s^*
0	1,1	0,5	4,33	3,91	3,46	2,50	0,28	9	0,64	1,56	0,013
1	1,25	0,60	4,71	2,83	2,75	2,30	0,32	9	0,65	1,85	0,017
2	1,45	0,77	5,10	1,77	2,02	2,09	0,35	9	0,70	2,20	0,024
3	1,67	0,99	5,49	0,84	1,32	1,89	0,36	9	0,77	2,57	0,034
4	1,91	1,27	5,86	0,06	0,62	1,67	0,36	9	0,86	2,96	0,068
5	2,23	1,72	6,31	-0,22	-0,22	1,38	0,36	9	1,00	3,46	0,11
6	2,59	2,32	6,77	-0,53	-1,14	1,01	0,34	9	1,15	4,03	0,21
7	3,04	3,21	7,27	0,54	-2,10	0,52	0,32	8	1,36	4,73	0,54
8	3,69	4,77	7,86	5,41	-2,71	-0,20	0,25	8	1,67	5,70	1,21
9	4,07	6,20	8,07	16,35	-0,22	-0,14	0,02	8	2,02	6,12	0,88
10	4,005	6,02	7,994	16,15	0,04	-0,006	0,003	8	2,005	6,005	1,11
11	4,0000	6,0002	7,9997	16,004	0,002	0,0003	0,0001	8	2,0001	5,9999	1,06
12	4,0000	6,0000	8,0000	16,0000	0,0000	0,0000	0,0000		2,0000	6,0000	

для перехода от точки α^v к точке α^{v+1} : четыре интегрирования «стоит» вычисление матрицы Z_{α} , остальные связаны с подбором шага процесса s^* ; эта величина также представлена в таблице. Таким образом, практически точное решение задачи стоит ~ 100 интегрирований задачи Коши. По таблице хорошо видно, что наибольшие трудности связаны с получением правильного значения $x^3(T)$; ради этого оказалось разумным иногда идти на удаление $x^1(T)$, $x^2(T)$ от нужных значений; однако эти отклонения легко (см. систему (8)) исправляются сравнительно малыми вариациями $\delta\alpha$.

На этой же задаче можно показать характерные трудности, связанные с выбором начального приближения α^0 . Выше уже отмечалось, что на значения $x(T)$ влияют не непосредственно значения α , а положения нулей функции $\psi_3(t)$, последние в свою очередь зависят от α_2 , α_3 . Однако в трехмерном пространстве $\{\alpha\}$ существует такая телесная (т. е. содержащая внутренние точки) область A , что при $\alpha \in A$ на отрезке $0 \leq t \leq 1$ ψ_3 имеет лишь один нуль (или даже ни одного). Изменению α в этой области A соответствует семейство решений П-системы, определяемое лишь двумя существенными параметрами: $\alpha_1 = T$ и положением нуля $\psi_3(t)$ (а может быть, даже одним T , если нуля нет). Таким образом, в области A отображение $Z(\alpha)$ вырождается: трехмерная область A отображается в двумерное (или даже одномерное) многообразие $Z(A)$.

В этом случае Z_{α}^{-1} не существует, хотя в расчетах «авоста» обычно не возникает: вычислительные ошибки в Z_{α} приводят к неравенству $\det(Z_{\alpha}) \neq 0$, однако решение $\delta\alpha = -Z_{\alpha}^{-1}(Z - X)$ становится, в сущности, случайным.

Этой случайности, разумеется, не следует доверять. Поэтому в расчетах была использована регуляризация — вместо системы $Z_{\alpha} \delta\alpha = X - Z$ решалась другая, а именно:

$$(Z_{\alpha}^* Z_{\alpha} + \varepsilon E) \delta\alpha = Z_{\alpha}^* (X - Z), \quad (9)$$

ε — величина очень малая по сравнению с элементами матрицы Z_{α}^*).

Первые шаги процесса поиска решения $Z(\alpha) = X$, начинающегося в точке $\alpha \in A$, проходят достаточно эффективно, однако если при этом точка α не выходит за пределы A , т. е. если не появляется второй нуль у $\psi_3(t)$, а он обычно не появляется, процесс находит в образе $Z(A)$ точку, ближайшую к X , после чего поиск застревает, локальными изменениями α нельзя сдвинуть точку $Z(\alpha)$ в сторону X .

*) Если $\det(Z_{\alpha})$ существенно отличен от нуля, решение (9) мало отличается от решения исходной системы (8). Если же $\det(Z_{\alpha}) \approx 0$, решение $\delta\alpha$ регуляризованной системы минимизирует квадратичную форму

$$(Z_{\alpha}^* \delta\alpha + Z - X, Z_{\alpha}^* \delta\alpha + Z - X) + \varepsilon (\delta\alpha, \delta\alpha).$$

В расчетах Дубовицкого и Рубцова выпуклая аппроксимация имела и еще одну важную цель — преодоление обсуждаемой вырожденности отображения $Z(\alpha)$ в области A . Для системы (2) образ $Z(A)$ уже не является двумерным, введение дополнительных компонент превращает его в трехмерный; однако в силу малого искажения первоначальной постановки задачи это очень «плоская» трехмерная область, схематически показанная на рис. 22. Очень маленьким перемещениям точки Z в направлении n (см. рис. 1) соответствуют очень большие перемещения в области A .

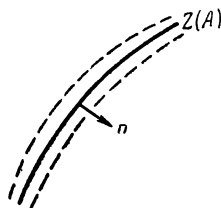


Рис. 22.

Сглаженное отображение $Z(\alpha)$ отличается сильной нелинейностью, и вывести точку α из области, где $\psi_z(t)$ имеет менее двух нулей, оказывается очень трудно. Хотя опубликованные в [30] результаты не позволяют восстановить характер затруднений, видно, что они значительны. Поэтому в [30] используется еще один прием — сначала задача решается при сильном сглаживании (коэффициент A в (4) порядка 1), полученное решение дает начальные данные для решения задачи с несколько большим значением A , и т. д. вплоть до достаточно больших A .

§ 28. Задача о вертикальном подъеме ракеты-зонда. Нелинейная П-система

Рассмотрим следующую вариационную задачу: движение управляемой системы описывается уравнениями

$$\begin{aligned} \dot{x}^1 &= -u, \\ \dot{x}^2 &= x^3, \\ \dot{x}^3 &= -g + \frac{1}{x^1(t)} [Vu - Ce^{-\gamma x^2(t)} [x^3(t)]^2], \\ 0 \leq t \leq T = 100; \quad 0 \leq u(t) \leq 0,04 \quad (u \in U), \\ \Gamma(x) &= 0: \quad x^1(0) = 1; \quad x^2(0) = 0; \quad x^3(0) = 0. \end{aligned} \quad (1)$$

Содержательный смысл (впрочем, здесь нас не очень интересующий) этой системы таков: $x^1(t)$ — есть переменная масса, $x^2(t)$ — вертикальная координата (высота), $x^3(t)$ — вертикальная скорость; V — постоянная, характеризующая величину реактивной тяги, g , C , γ — заданные постоянные, связанные с силой тяготения аэродинамическим сопротивлением и убыванием плотности воздуха с высотой.

Задача решалась при значениях (см. [99])

$$g = 0,01, \quad V = 2,0, \quad C = 0,05, \quad \gamma = 0,01.$$

Управление $u(t)$ определяет режим расхода горючего, ограничение $u \geq 0$ имеет очевидный физический смысл, ограничение $u \leq 0,04$ носит технический характер.

Вариационная задача (см. [99]) состоит в определении управления $u(\cdot)$ так, чтобы обеспечить

$$\max_{u(\cdot)} x^2(T) \quad (F_0[u(\cdot)] \equiv x^2(T))$$

при дополнительном условии

$$x^1(T) = 0,2 \quad (F_1[u(\cdot)] \equiv x^1(T) - 0,2)$$

(максимальная высота в заданное время при заданном запасе горючего).

Задача эта подробно исследовалась как аналитическими методами, так и численно. Решение ее хорошо известно; точнее, известна качественная структура решения, что позволяет построить простые методы его приближенного определения.

Структура решения такова:

$$u(t) = \begin{cases} 0,04 & \text{при } 0 \leq t < t_1, \\ u^*(t) & \text{при } t_1 < t < t_2, \\ 0 & \text{при } t_2 < t \leq T. \end{cases}$$

Эта структура содержит два параметра t_1, t_2 , для функции $u^*(t)$ можно написать некоторое уравнение (аналог уравнения Эйлера), допускающее численное интегрирование; t_1, t_2 подбираются из условия $x^2(T) = 0,2$ и условия $\max x^2(T)$ или из соответствующего условия трансверсальности. Мы используем эту задачу в качестве теста и проиллюстрируем характерные трудности, возникающие при решении нелинейных П-систем.

Будем решать задачу методом § 14. Для этого выпишем систему уравнений для $\psi(t)$:

$$\begin{aligned} -\dot{\psi}_1 &= -\frac{1}{(x^1)^2} [Vu - Ce^{-\gamma x^2} (x^3)^2] \psi_3, \\ -\dot{\psi}_2 &= \frac{C\gamma}{x^1} e^{-\gamma x^2} (x^3)^2 \psi_3, \\ -\dot{\psi}_3 &= \psi_2 - 2 \frac{Ce^{-\gamma x^2}}{x^1} x^3 \psi_3, \end{aligned} \quad (2)$$

и уравнение принципа максимума, которое, поскольку u входит в задачу линейно, имеет вид

$$\begin{aligned} H_0[x(t), \psi(t)] + u(t) H_1[x(t), \psi(t)] &= \\ &= \max_u \{H_0[x(t), \psi(t)] + u H_1[x(t), \psi(t)]\}. \end{aligned} \quad (3)$$

Точный вид функций H_0, H_1 не очень важен.

Решение этого уравнения просто:

$$u = \begin{cases} 0 & \text{при } H_1 < 0, \\ 0,04 & \text{при } H_1 > 0, \\ 0 \leq u \leq 0,04 & \text{при } H_1 = 0, \end{cases}$$

однако случай $H_1=0$ связан, как мы увидим, с существенными трудностями. Итак, задав некоторые начальные значения $\psi_2(0)$ и $\psi_3(0)$ ($\psi_1(0)$, фиксируем, положив $\psi_1(0)=1$; следует, конечно, иметь в виду и вариант $\psi_1(0)=-1$), проинтегрируем П-систему. Таким образом, формально речь идет о решении двух уравнений с двумя неизвестными $\psi_2(0)$ и $\psi_3(0)$:

- 1) $x^1(T)=0,2$ (дополнительное условие задачи);
- 2) $\psi_3(T)=0$ (условие трансверсальности),

причем $x^1(T)$ и $\psi_3(T)$ определяются через $\psi_2(0)$, $\psi_3(0)$ решением задачи Коши для П-системы (1)–(3). Что же получается при реализации этой программы на ЭВМ? Следующее ниже основано на проводившихся автором вычислениях и является изложением экспериментальных фактов. Прежде всего отметим важные для дальнейшего свойства функции $H_1(x, \psi)$.

1. Функция $\dot{H}_1[x(t), \psi(t)] \equiv \frac{\partial H_1}{\partial x} \frac{dx}{dt} + \frac{\partial H_1}{\partial \psi} \frac{d\psi}{dt}$ после замены \dot{x} и $\dot{\psi}$ правыми частями соответствующих уравнений не содержит явной зависимости от u . Этот факт проверяется прямыми выкладками, которые мы в силу их простоты и громоздкости опускаем.

2. Функция $\dot{H}[x(t), \psi(t), u(t)] \equiv \frac{\partial H_1}{\partial x} \frac{dx}{dt} + \frac{\partial H_1}{\partial \psi} \frac{d\psi}{dt}$, вычисленная аналогичным образом, явно зависит от u , причем линейно, поскольку u входит линейно в правые части уравнений (1), (2).

Итак, для почти всех точек $\{\psi_2(0), \psi_3(0)\}$ решение задачи Коши для П-системы оказывается единственным; если на траектории и встречается ситуация $H_1[x(t^*), \psi(t^*)]=0$, при которой значение u неоднозначно, она носит, так сказать, мгновенный характер.

Если $\dot{H}_1[t^*-0] < 0$, то при любом выборе $u(t^*)$, в силу независимости H_1 от u (явно), \dot{H}_1 меняется непрерывно, $\dot{H}_1 < 0$ в окрестности точки t^* , и траектория П-системы продолжается однозначно.

Описанные выше траектории П-системы гладко зависят от $\{\psi_2(0), \psi_3(0)\}$, но в действительности они образуют не двух-, а лишь однопараметрическое семейство: $\{\psi_2(0), \psi_3(0)\}$ однозначно определяют один параметр t^* , а он в свою очередь однозначно определяет $u(t)$ и траекторию x . Однако в плоскости $\{\psi_2(0), \psi_3(0)\}$ есть особая линия L , и если $\{\psi_2(0), \psi_3(0)\} \in L$, решение П-системы имеет иной характер: сначала $H_1[x(t), \psi(t)] > 0$ и $u(t)=0,04$;

с ростом времени $H_1[t]$ убывает, и в некоторый момент t^* , зависящий от положения $\psi(0)$ на L , в нуль одновременно обращаются обе функции: $H_1[x(t^*), \psi(t^*)] = \dot{H}_1[x(t^*), \psi(t^*)] = 0$. И теперь оказывается возможным продолжить траекторию Π -системы тремя разными способами.

Обозначим через $\tilde{u}(t)$ решение линейного по u уравнения $\dot{H}[x(t), \psi(t), u] = 0$. Пусть $0 < \tilde{u} < 0,04$ (а это так и есть, по крайней мере для некоторого участка L). Различные продолжения траектории Π -системы определяются выбором управления при $t > t^*$.

I. $u(t) = 0,04$. В этом случае $\dot{H} > 0$, $H_1[x(t), \psi(t)]$, коснувшись оси $H_1 = 0$ в точке t^* , остается положительной, и принцип максимума выполнен.

II. $u(t) = 0$. В этом случае $\dot{H} < 0$, $H_1[x(t), \psi(t)]$, коснувшись оси $H_1 = 0$, переходит от положительных значений (при $t < t^*$) к отрицательным; принцип максимума выполнен и на этом продолжении.

При численном интегрировании, если не принять специальных мер, реализуются только эти два случая, так как чистый нуль в H_1 получить нельзя. Однако среди этих «однозначных» траекторий нет искомой оптимальной. Кстати, мы по-прежнему не получили двухпараметрического семейства траекторий, и попытка выбором $\{\psi_2(0), \psi_3(0)\}$ удовлетворить два условия при $t = T$ обречена на неудачу. В расчетах это проявится в виде нерегулярного, непредсказуемого изменения величин $x^1(T)$ и $\psi_3(T)$ при очень малых изменениях $\psi(0)$. Такие явления отмечались в литературе, например, в [57] (стр. 88), где они объясняются неустойчивостью. Термин «неустойчивость» достаточно широк, а примеры, в которых эта неустойчивость была обнаружена, к сожалению, не приводятся. Подробный анализ таких примеров в высшей степени важен, так как необходимо четко выявить и классифицировать различные причины неудач, с тем чтобы можно было сознательно разрабатывать соответствующие усовершенствования алгоритмов.

На рисунке 23 показана характерная эволюция функции $H_1[x(t), \psi(t)]$ на двух траекториях, определяемых значениями:

- 1) $\psi_2(0) = -0,025$; $\psi_3(0) = 0,57504$; $H_1[0] = 0,15010$,
- 2) $\psi_2(0) = -0,025$; $\psi_3(0) = 0,57505$; $H_1[0] = 0,15008$.

III. Основное продолжение: при $t > t^*$ управление $\tilde{u}(t)$ определяется уравнением $\dot{H}[x(t), \psi(t), u] = 0$. Принцип максимума выполнен, так как из $\dot{H}[t] = 0$ следует $\dot{H}_1[t] = 0$ и $H_1[t] = 0$. Разумеется, этот особый режим возможен лишь при $\tilde{u} \in [0; 0,04]$. В любой момент $t^{**} \geq t^*$ траектория Π -системы может быть переведена в режим I ($u(t) = 0,04$ при $t > t^{**}$) или II ($u(t) = 0$ при $t > t^{**}$). Итак, решение задачи Коши для Π -системы оказалось

существенно неоднозначным, каждой точке $\phi(0) \in L$ соответствуют два однопараметрических (с параметром t^{**}) семейства траекторий. Параметризуя положение $\phi(0)$ на L , для чего наиболее удобным является параметр t^* , получаем двухпараметрические семейства траекторий Π -системы. В одном из них, а именно во втором, и находится оптимальная траектория. Догадаться об этом можно и по численным результатам, дающим достаточно грубое (в смысле u) приближенное решение (см. § 29, рис. 25). Теперь уже можно,

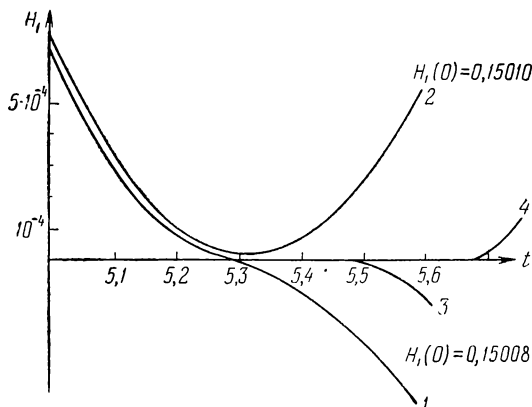


Рис. 23.

используя условия при $t=T$, найти параметры t^* , t^{**} , а тем самым, и оптимальное управление. Делается это, например, так:

1. Зададим некоторое значение параметра t^* .
2. Проинтегрируем на $[0, t^*]$ систему $\dot{x}=f(x, u)$, положив $u(t)=0,04$; это соответствует тому, что на $[0, t^*]$ будем иметь $H_1 > 0$. Итак, $x(t)$ определена на $[0, t^*]$.
3. Из уравнений $H_1[x(t^*), \phi]=0$; $\dot{H}_1[x(t^*), \phi]=0$ найдем $\phi_2(t^*)$ и $\phi_3(t^*)$ положив, например, $\phi_1=1$. (При желании можно проинтегрировать от t^* к $t=0$ систему для $\phi(t)$ вдоль известной уже $x(t)$, и получить точку $\{\phi_2(0)/\phi_1(0), \phi_3(0)/\phi_1(0)\} \in L$, соответствующую параметру t^* .)

Кстати, заметим, что для значений t^* получаем естественный интервал: $t^* \in (0; 1/0,04)$, поскольку лишь при $x^1(t) > 0$ система уравнений может быть физически осмыслена.

4. Имея теперь данные Коши для Π -системы при $t=t^*$, интегрируем ее, определяя $u(t)$ из уравнения $\ddot{H}[x(t), \phi(t), u(t)]=0$ до момента t^{**} , он определяется дополнительным условием задачи $x^1(t^{**})=0,2$ (таким образом, для t^* имеем дополнительное ограничение $t^* \leq 0,8/0,04$); нетрудно видеть, что $x^1(t)$ монотонно убывает вдоль траектории $x(t)$ (если $u > 0$, что и выполняется при $0 \leq t \leq t^{**}$).

Далее траектория однозначно продолжается при $u(t) \equiv 0$, что принципу максимума не противоречит, так как автоматически приводит к $H_1 < 0$.

Итак, построено однопараметрическое (с параметром t^*) семейство решений П-системы, удовлетворяющих дополнительному условию задачи $x^1(T) = 0, 2$. На этом семействе $x^2(T)$ становится функцией одного параметра t^* , функцией весьма гладкой, и нахождение

$$\max_{t^*} x^2(T)$$

осуществляется без труда.

Итак, решение задачи оптимального управления найдено по схеме, формально совпадающей с той, с которой все началось: строится двухпараметрическое семейство решений П-системы, и значения параметров определяются из дополнительного условия и условия максимума $x^2(T)$. Однако разница, и очень существенная, состоит в том, что в качестве этих параметров не удастся взять данные Коши $\phi(0)$. Реализация же всей программы потребовала привлечения достаточно подробных предварительных сведений о качественном характере искомого решения.

§ 29. Задача о вертикальном подъеме ракеты

Это была, видимо, одна из первых задач оптимального управления, подробно исследованная аналитически и неоднократно решавшаяся численно. В частности, и автор использовал ее в качестве теста в методических расчетах. Уравнения движения:

$$\dot{x}^1 = -u; \quad \dot{x}^2 = x^3; \quad \dot{x}^3 = -g + [Vu - Q(x)]/x^1; \quad x(0) = \{1; 0; 0\}, \quad (1)$$

где $g = 0,981 \cdot 10^{-4}$; $V = 0,0168$; $Q(x) = 0,96 (x^3)^2 \exp(-14,7x^2)$. Вводятся функционалы

$$F_0[u(\cdot), T] \equiv 1 - x^1(T); \quad F_1[u(\cdot), T] \equiv x^2(T), \quad (2)$$

и ставится задача: найти управление $\{u(\cdot), T\}$, доставляющее

$$\min_{u(\cdot), T} F_0[u(\cdot), T] \quad (3)$$

при $F_1[u(\cdot), T] = 1,49, 0 \leq u(t) \leq U^*, 0 \leq t \leq T$. Решение задачи хорошо известно (см. [46]):

$$u(t) = \begin{cases} U^* & \text{при } 0 < t < t_1, \\ u^*(t) & \text{при } t_1 < t < t_2, \\ 0 & \text{при } t_2 < t < T. \end{cases} \quad (4)$$

Параметры t_1, t_2, T и функция $u^*(t)$ на участке «особого режима» (t_1, t_2) могут быть рассчитаны с высокой степенью точности

(см., например, § 28). В решении этой задачи

$$F_0 = \int_0^T u(t) dt \approx 0,68; \quad \int_0^{t_1} u(t) dt = U^+ t_1 \approx 0,33.$$

Иногда рассматривается задача с $U^+ = \infty$. В этом случае начальный участок решения вырождается в δ -функцию с интенсивностью $\sim 0,33$ и полюсом при $t=0^*$). В настоящее время интерес представляет не само решение, а процесс его получения, характерные трудности, их анализ и способы преодоления. Поэтому здесь анализируется процесс решения задачи в том виде, каким он был в 1962 г., хотя, повторив расчеты по нынешним программам, автор мог бы представить более эффектные результаты.

Схема вычисления. Вводилась сетка $0 = t_0 < t < t_2 < \dots < t_N$ и кусочно постоянное управление $u_{n+1/2}$. Сетка была неравномерной, более густой на интервале $[0; 20]$, $N \approx 100$. Система уравнений (1) интегрировалась с шагом $dt = 0,1(t_{n+1} - t_n)$ методом Рунге—Кутты второго порядка точности. При этом запоминались значения $x\left(\frac{t_n + t_{n+1}}{2}\right)$. Интегрирование велось до тех пор, пока $x^2(t) > 0$. Таким образом, для определения T использовалось уравнение $x^2(T) = 0$. Это есть не что иное, как условие трансверсальности по параметру T , так как при $t \approx T$ $u(t) = 0$; $\frac{\partial F_0}{\partial T} = 0$; $\frac{\partial F_1}{\partial T} = x^2(T)$. Затем интегрировалась сопряженная система с данными Коши $\psi(T) = \{0; 1; 0\}$, интегрирование велось грубо, с шагом $dt = (t_{n+1} - t_n)$. Зная $\psi(t)$, можно записать формулу для производной

$$\delta F_1[\delta u(\cdot)] = \int_0^T \psi(t) f_u[t] \delta u(t) dt = \sum_{n=0}^{N-1} \delta u_{n+1/2} \int_{t_n}^{t_{n+1}} \psi f_u dt. \quad (5)$$

Результаты запоминались в виде таблицы чисел $h_{n+1/2}^1$:

$$h_{n+1/2}^1 = \int_{t_n}^{t_{n+1}} \psi f_u dt \approx \frac{1}{2}(\psi_{n+1} + \psi_n, f_u[t_{n+1/2}]) (t_{n+1} - t_n),$$

$$h_{n+1/2}^0 = t_{n+1} - t_n, \quad \text{т. к.} \quad \delta F_0 = \int_0^T \delta u dt. \quad (6)$$

*) Так как в уравнение (1) входит член $u/x^1(t)$, а $x^1(t)$ рвется в полюс δ -функции, расчет скачка $x(t)$ при переходе через полюс δ -функции нетривиален и, если пренебречь сопротивлением воздуха, осуществляется по известной формуле Циолковского. В расчетах необходимая точность обеспечивается иначе, за счет того, что шаг численного интегрирования (1) меньше шага сетки для u .

Назначались числа s^- , $s_{n+1/2}^+$, ограничивающие величины вариаций $s_{n+1/2}^- \leq \delta u_{n+1/2} \leq s_{n+1/2}^+$, а вариация управления определялась решением простой задачи линейного программирования. Далее находилось новое управление $u_{n+1/2} + \delta u_{n+1/2}$, и так далее. Выше были опущены некоторые детали, которые удобнее будет объяснить в комментариях к результатам.

Первый расчет представлен в табл. 1 функциями u при $v=0,5$, 10, 15, 20 (v — номер итерации) и соответствующими значениями F_0 . Условие $F_1=1,49$ было выполнено с точностью $\approx 0,0005$. На 20-й итерации получено значение $F_0=0,68022$, на 0,0017 больше точного $\min F_0$. На первый взгляд это неплохо, точность $\approx 0,25\%$. Однако в действительности результат не очень хороший. Ведь естественнее относить ошибку 0,0017 не к F_0 , а к выигрышу в F_0 по сравнению с тривиальным управлением, взятым в качестве исходного. Этот выигрыш $\approx 0,02$, и теперь точность расчета $\approx 10\%$. Легко указать недостаток численного решения — плохо выражена δ -функция в $u(t)$, она сильно «размазана», причем продолжение расчета не приводило к улучшению.

Второй и третий расчеты проводились так же, как и первый, но при другой начальной функции $u(t)$, содержащей «подсказку» — численный аналог δ -функции, причем третий расчет начинался даже с точного решения. В обоих случаях процесс итераций сопровождался ухудшением траектории: значение F_0 повышалось, а не понижалось (обычно в таком случае следует возвращаться к исходному управлению и, например, варьировать его с меньшим шагом δu ; здесь этот механизм был отключен). В чем же дело? Возможны две причины неправильной работы алгоритма: либо слишком велик шаг δu , и сказываются неучтенные при вычислении δF погрешности $O(\|\delta u\|^2)$, либо неточно вычисляется производная функционала (из-за ошибок численного интегрирования или из-за ошибок в программе). Так как уменьшение δu (s^- , s^+) не привело к улучшению, стало ясно, что дело в грубости вычисления величин $h_{n+1/2}^1$ по формуле (6). Нетрудно было также догадаться, что ошибка, в сущности, велика лишь на первом счетном интервале: ведь на (t_0, t_1) расходуется $\sim 0,33$ массы, все величины резко изменяются. Было внесено только уточнение расчета $h_{1/2}^1$: первый интервал сетки (t_0, t_1) был разбит на 10 частей, что позволило более точно интегрировать систему для ϕ и более точно вычислять интеграл в (6).

Четвертый расчет, результат которого представлен в табл. 2, привел уже к четкому образованию численного аналога δ -функции, причем $F_0 - \min F_0 \approx 0,0001$. Это потребовало 50—60 итераций.

П р и н ц и п м а к с и м у м а. Каждый расчет заканчивается обычно в ситуации, когда значение функционала F_0 практически стабилизируется. Это может быть следствием достижения минимума или следствием неэффективности метода минимиза-

Т а б л и ц а 1

F_0	0,70000	0,68315	0,68198	0,68092	0,68022	0,6785
t	$v = 0$	$v = 5$	$v = 10$	$v = 15$	$v = 20$	точное и (t)
0	0,1000	0,0776	0,0992	0,1396	0,1926	0,6560
0,5	0,1000	0,0776	0,0992	0,1396	0,1399	0,0157
1,0	0,1000	0,0776	0,0992	0,1299	0,1033	0,0157
1,5	0,1000	0,0776	0,0883	0,0939	0,0792	0,0157
2,0	0,1000	0,0776	0,0864	0,0704	0,0656	0,0158
2,5	0,1000	0,0776	0,0610	0,0551	0,0513	0,0158
3,0	0,1000	0,0776	0,0602	0,0468	0,0397	0,0159
3,5	0,1000	0,0776	0,0602	0,0398	0,0355	0,0159
4,0	0,1000	0,0721	0,0441	0,0345	0,0309	0,0160
4,5	0,1000	0,0721	0,0441	0,0222	0,0239	0,0161
5,0	0,1000	0,0721	0,0352	0,0246	0,0220	0,0162
5,5	0,1000	0,0721	0,0352	0,0191	0,0197	0,0164
6,0	0,1000	0,0721	0,0352	0,0178	0,0175	0,0166
7,0		0	0,0070	0,0200	0,0190	0,0170
8,0		0	0,0200	0,0197	0,0210	0,0174
9,0		0,0077	0,0150	0,0170	0,0202	0,0178
10,0		0,0135	0,0205	0,0221	0,0210	0,0183
11,0		0,0151	0,0206	0,0190	0,0202	0,0188
12,0		0,0172	0,0212	0,0200	0,0196	0,0192
13,0		0,0150	0,0200	0,0213	0,0207	0,0197
14,0		0,0210	0,0178	0,0192	0,0188	0,0202
15,0		0,0231	0,0182	0,0198	0,0198	0,0208
16,0		0,0190	0,0180	0,0175	0,0185	0,0214
17,0		0,0190	0,0175	0,0175	0,0173	0,0220
18,0		0,0113	0,0155	0,0150	0,0163	0,0226

Т а б л и ц а 2

F_0	II				III		IV	
		0,6789	0,6795	0,6800	0,6785	0,6795	0,7000	0,6786
t	$\nu = 0$	ν_1	ν_2	ν_3	$\nu = 0$	ν_1	$\nu = 0$	ν_1
0,5	0,6000	0,4841	0,3050	0,1977	0,6560	0,3623	0,1000	0,7856
1,0	0,1000	0,0547	0,1128	0,1434	0,0157	0,0635	0,1000	0
1,5	0,1000	0,0448	0,0734	0,1051	0,0157	0,0547	0,1000	0,0012
2,0	0,1000	0,0369	0,0609	0,0782	0,0157	0,0551	0,1000	0,0060
2,5	0,1000	0,0297	0,0441	0,0583	0,0158	0,0514	0,1000	0,0084
3,0	0,1000	0,0287	0,0327	0,0500	0,0158	0,0352	0,1000	0,0104
3,5	0,1000	0,0263	0,0303	0,0400	0,0159	0,0253	0,1000	0,0127
4,0	0,1000	0,0232	0,0358	0,0343	0,0159	0,0328	0,1000	0,0138
4,5	0	0,0240	0,0279	0,0316	0,0160	0,0347	0,1000	0,0149
5,0	0	0,0216	0,0235	0,0246	0,0161	0,0233	0,1000	0,0154
5,5	0	0,0201	0,0232	0,0226	0,0162	0,0251	0,1000	0,0172
6,0	0	0,0206	0,0237	0,0212	0,0164	0,0215	0,1000	0,0171
7	0	0,0199	0,0211	0,0209	0,0166	0,0201	0,1000	0,0183
8	0	0,0197	0,0206	0,0198	0,0170	0,0203	0	0,0185
9	0	0,0197	0,0202	0,0199	0,0174	0,0210	0	0,0190
10	0	0,0201	0,0209	0,0207	0,0178	0,0220	0	0,0192
11	0	0,0199	0,0208	0,0209	0,0183	0,0208	0	0,0193
12	0	0,0189	0,0206	0,0201	0,0188	0,0206	0	0,0193
13	0	0,0189	0,0199	0,0199	0,0192	0,0203	0	0,0193
14	0	0,0196	0,0198	0,0202	0,0197	0,0196	0	0,0186
15	0	0,0195	0,0194	0,0186	0,0202	0,0190	0	0,0179
16	0	0,0176	0,0186	0,0180	0,0208	0,0173	0	0,0160
17	0	0,0162	0,0176	0,0165	0,0214	0,0155	0	0,01446
18	0	0,0142	0,0163	0,0010	0,0220	0,0130	0	0,0123
	0	0,0125	0,0143		0,0226	0,0120		

ции. В данном случае в вопросе можно легко разобраться, построив и проанализировав конус достижимости. Начнем с конуса допустимых вариаций K_u . Пусть получено решение типа (4). Тогда K_u имеет простую структуру

$$\delta u(t) \in K_u: \begin{cases} \delta u \leq 0 & \text{при } u(t) = U^+, \\ \delta u \geq 0 & \text{при } u(t) = 0, \\ \text{произвольная} & \text{при } 0 < u(t) < U^+. \end{cases} \quad (7)$$

Конус K_u отображается в двумерное пространство $\{\delta F_0, \delta F_1\}$

$$\begin{aligned} \delta F_0 &= \int_0^T \delta u(t) dt, \\ \delta F_1 &= \int_0^T w_1(t) \delta u(t) dt. \end{aligned} \quad (8)$$

Множество вариаций $\delta u(t)$ на той части $[0, T]$, где $0 < u < U^+$ (а в расчетах I—IV $U^+ = \infty$), образует линейное пространство, следовательно, и его образ в конусе достижимости тоже есть линейное пространство. Для того чтобы траектория была оптимальной, это последнее линейное пространство не должно совпадать с плоскостью. Значит, оно должно вырождаться в прямую, что возможно лишь при $w_1(t) = \text{const}$ при $u(t) > 0$. В ситуациях, в которых прекращались итерации, постоянство $w_1(t)$ на активном участке траектории ($u > 0$) выполнялось с четырьмя знаками. На пассивном участке ($u = 0$), где $\delta u \geq 0$, выполнялось, как и должно быть, неравенство $w_1(t) < C$. На рис. 24 показана структура конуса достижимости: нанесены векторы $\{1, w_1(t)\}$ для разных t .

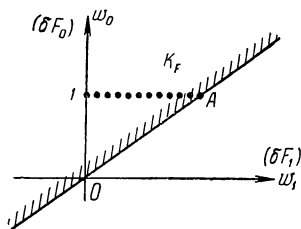


Рис. 24.

Все векторы, соответствующие активному участку траектории, представлены одной точкой A , остальные — разными. Таким образом, конус смещений K_F есть полуплоскость, лежащая выше прямой OA . Дальнейшие эксперименты имели целью иллюстрировать эффект некоторых несложных вычислительных приемов. Они проводились позже, по другой программе, отличающейся в основном, большим числом N (≈ 500 , и из них ≈ 200 на активном участке).

Конструкция окрестности $\delta U(t)$, определяемая в данном случае заданием чисел $s_{n+1/2}^-$, $s_{n+1/2}^+$, кроме всего прочего, имеет целью разрешить большую вариацию там, где она наиболее «выгодна».

Это особенно важно в тех задачах, где процесс малых вариаций должен привести к численному аналогу δ -функции. Здесь был использован следующий прием. Определение $\delta u_{n+1/2}$ решением задачи

$$\begin{aligned} \min_{\{\delta u\}} \sum_{n=0}^{N-1} h_{n+1/2}^0 \delta u_{n+1/2} \quad (h^0 \equiv 1), \\ F_1 + \sum_{n=0}^{N-1} h_{n+1/2}^1 \delta u_{n+1/2} = 0, \\ s_{n+1/2}^- \leq \delta u_{n+1/2} \leq s_{n+1/2}^+, \end{aligned} \quad (9)$$

связано с вычислением вектора $\{1, g\}$, причем

$$\delta u_{n+1/2} = \begin{cases} s_{n+1/2}^- & \text{при } H_{n+1/2}^0 > 0, \\ s_{n+1/2}^+ & \text{при } H_{n+1/2}^0 < 0, \end{cases}$$

где $H_{n+1/2}^0 = h_{n+1/2}^1 + g h_{n+1/2}$ характеризует в некотором смысле эффективность вариации управления на интервале (t_n, t_{n+1}) : чем H^0 больше, тем выгоднее варьировать u именно на этом счетном интервале. На тех же интервалах сетки, где $H^0 \approx 0$, можно брать любые вариации $\delta u_{n+1/2}$, лишь бы сохранялось соотношение $F_1 + \Sigma h^1 \delta u \approx 0$. Поэтому для следующего шага процесса величины s^\pm рассчитывались по формулам

$$\begin{aligned} s_{n+1/2}^+ &= S + \xi |H_{n+1/2}^0| / \|h_{n+1/2}\|, \\ s_{n+1/2}^- &= -S - \xi |H_{n+1/2}^0| / \|h_{n+1/2}\|, \end{aligned}$$

а величина ξ определялась соотношением типа $\frac{1}{N} \sum s_{n+1/2}^+ \approx 3S \div 5S$.

Величина S , таким образом, и определяла шаг процесса. Разумеется, числа s^\pm корректировались затем учетом условий $0 \leq u + s^-, u + s^+ \leq U^+$. В §§ 34, 35 описана другая техника учета эффективности вариации на данном интервале. В табл. 3 представлены расчеты (I—IV), отличающиеся только тем, что в II величины s^-, s^+ вычислялись так, как описано выше, а в IV $s^\pm = \text{const}$. Видно, что расчет II примерно в три раза быстрее. Нужно, однако, подчеркнуть, что это прежде всего связано с наличием δ -функции в искомом оптимальном управлении. В тех задачах, где таких (или аналогичных) особенностей нет, эффект значительно слабее.

Регулирование шага S осуществлялось простым механизмом обратной связи: после определения δu вычислялось предсказание

$$\delta F_1 = \sum_{n=0}^{N-1} h_{n+1/2}^1 \delta u_{n+1/2}.$$

Т а б л и ц а 3

v	I			II		III		IV	
	$x^1(T)$	$x^2(T)$	S	$x^1(T)$	S	$x^1(T)$	S	$x^1(T)$	$x_2(T)$
0	0,7000		0,015	0,7000	0,15	0,7000	1,5	0,700000	1,676
6	0,68869	1,491150	0,007	0,68517	0,11	0,68033	0,27	0,6883	1,48999
12	0,687794	1,48998	0,025	0,68298	0,15	0,67981	0,13	0,6865	1,48986
18	0,68653	1,48970	0,086	0,68109	0,15	0,67935	0,16	0,6834	1,48979
24	0,68425	1,48993	0,18	0,68001	0,15	0,67904	0,13	0,6815	1,48954
30	0,68166	1,48941	0,14	0,67953	0,11	0,67895	0,10	0,6809	1,48950
36	0,68051	1,48989	0,18	0,67917	0,16	0,67884	0,21	0,6806	1,48975
42	0,67975	1,48895	0,14	0,67899	0,11	0,67879	0,17	0,6803	1,48973
48	0,67936	1,48943	0,11	0,67884	0,11	0,67875	0,13	0,6800	1,48983
54	0,67915	1,48988	0,15	0,67878	0,12	0,67870	0,10	0,6799	1,48974
60	0,67896	1,48942	0,11	0,67872	0,12	0,67868	0,13	0,6797	1,48985
66	0,67887	1,48981	0,15	0,67869	0,12	0,67864	0,17	0,6796	1,48963
72	0,67879	1,48963	0,19	0,67863	0,12	0,67865	0,13	0,6795	1,48962
78	0,67873	1,48955	0,09	0,67864	0,11	0,67861	0,11	0,6794	1,48957
84	0,67869	1,48940	0,12	0,67862	0,12	0,67863	0,12	0,6793	1,48980
90	0,67865	1,48951	0,09					0,6792	1,48960
96	0,67868	1,48975	0,14						

С новым управлением $u + \delta u$ интегрировалась система $\dot{x} = f$, определялись истинное приращение ΔF_1 и величина, характеризующая точность линейного приближения

$$\eta = |\delta F_1 - \Delta F_1| \left/ \sum_{n=0}^{N-1} |\delta u_{n+1/2} h_{n+1/2}^1| \right|$$

(см. в § 20 объяснение, почему нельзя брать отношение разности, например, к $|\Delta F_1|$). В зависимости от величины η происходил расчет S :

$$S = \begin{cases} 1,15S & \text{при } \eta < 0,25, \\ S & \text{при } 0,25 \leq \eta \leq 0,5, \\ 0,75S & \text{при } \eta > 0,5. \end{cases}$$

В табл. 3 представлены расчеты I, II, III, отличающиеся только величиной первоначально заданного S . Видно, как во всех расчетах достаточно быстро вырабатывается одна и та же величина $S \approx 0,1$. Табл. 1, 2 публиковались в [87], табл. 3 — в [89].

Сравнение с решением задачи методом проекции градиента было проведено для другого варианта, отличающегося следующим:

- 1) $u(t) \in U$: $0 \leq u(t) \leq 0,04$;
- 2) фиксировано $T = 100$;
- 3) $g = 0,01$, $V = 2,0$, $Q = 0,05(x^3)^2 e^{-0,1x^2}$;

4) задача: $\max x^2(T)$ при $x^1(T)=0,2$ (отличие параметров в основном связано с другой системой единиц).

Эти данные были взяты из работы [99], где задача решалась классическим вариантом метода проекции градиента, применимость которого обеспечивалась ликвидацией условия $u(t) \in U$ при помощи замены u на неограниченное v :

$$u(v) = \begin{cases} 0 & \text{при } v < 0, \\ 0,04(3v^2 - 2v^3) & \text{при } 0 \leq v \leq 1, \\ 0,04 & \text{при } v > 1 \end{cases}$$

(эта замена аналогична замене $u=0,02+0,02 \sin v$). Итерация в работе [99] состоит из следующих вычислений.

1. Имея некоторое $v(t)$, интегрируем систему $\dot{x}=f$.

2. На полученной траектории вычисляются производные функционалов $w_0(t)=\partial x^2(T)/\partial v(\cdot)$, $w_1(t)=\partial x^1(T)/\partial v(\cdot)$, что требует только одного интегрирования сопряженной системы $\left(w_1(t) \right.$ вычис-

ляется сразу, так как $x^1(T)=x^1(0)-\int_0^T u(t) dt$).

3. Определяется вариация $\delta v(t)$ решением задачи $\min_{\delta v} \int_0^T w_0(t) \delta v dt$

при условиях $\int w_1 \delta v dt = 0$; $\int \delta v^2 dt = 1$.

4. Находится новое управление $v(t) := v(t) + S \delta v(t)$.

5. Подбор шага S осуществляется следующим образом: с новым управлением интегрировалась система $\dot{x}=f$ и вычислялось фактическое приращение функционала $x^2(T)$. Если были выполнены условия: $|x^1(T)-0,2| \leq \epsilon$ и $\Delta x^2(T) > \frac{1}{2} \delta x^2(T)$ ($\delta x^2(T)$ — приращение, вычисленное по линейной теории), итерация считалась завершенной, и вычисления продолжались с пункта 2. Если $\delta x^2(T) < \frac{1}{2} \delta x^2(T)$, таким же образом испытывалось управление $v(t) + \frac{1}{2} S \delta v(t)$, и т. д. Если оказывалось $|x^1(T)-0,2| > \epsilon$, предусматривались итерации с целью погашения невязок в этом условии. Процесс решения показан в заимствованной из [99] табл. 4 (II) (правда, нет сведений о числе интегрирований системы $\dot{x}=f$ для установления S). В той же табл. 4 представлены и наши расчеты; величина $x^1(T)$ не показана, так как условие $x^1(T)=0,2$ выполнялось в силу его линейности по u (по v оно нелинейно) со всеми машинными знаками. В табл. 4 (I) представлены предсказанное

Т а б л и ц а 4

ν	I			ν	II	
	$x^2 (T)$	$\delta x^2 (T)$	$\Delta x^2 (T)$		$x^2 (T)$	$x^1 (T)$
0	54,701	23,4	26,6	0	54,760	0,19971
1	81,381	21,8	24,2	5	80,973	0,19902
2	105,61	15,2	16,1	10	103,026	0,19919
3	121,73	9,4	6,9	15	115,658	0,19957
4	129,61	5,3	2,5	20	122,070	0,19981
5	131,12	3,2	-1,7	25	127,679	0,19978
6	129,44	3,1	1,9	30	131,258	0,19973
7	131,32	1,4	0,54	34	132,177	0,19971
8	131,86	1,2	-0,36	40	132,343	0,19969
9	131,50	0,9	0,57	44	132,345	0,19966
10	132,07	0,29	0,07	50	132,346	0,19967
11	132,14	0,07	0,04			
12	132,18	0,0008				

приращение $\delta x^2 (T)$ и фактическое $\Delta x^2 (T)$. Обратите внимание на 5-ю и 8-ю итерации, когда вместо предсказанного увеличения $x^2 (T)$ происходило уменьшение. Обычно в таких ситуациях предпочитают вернуться к предыдущему управлению и варьировать его с уменьшенным шагом. Здесь это не делалось, только уменьшалось S . Видно, что можно поступать и так. В [99] получено несколько большее значение $x^2 (T)$ (на 0,1%), чем в наших расчетах. Однако это связано с некоторым «перерасходом» массы: $x^1 (T)$ меньше заданного (на 0,16%). Таким образом, в наших расчетах поиск экстремума протекал примерно в три раза быстрее. С чем это связано? Обычно эффективность процесса оптимизации связывают с качеством направления изменения аргумента δu . Здесь это едва ли так. На рис. 25 показаны последовательно получаемые *) в наших расчетах функции $u(t)$ (расчет, как и в [99], начинался с $u(t) \equiv 0,008$). Аналогичный рисунок приведен и в [99]. Сравнение показывает, что в обоих расчетах $u(t)$ прошло примерно одну и ту же последовательность функций. Так, $u(t)$ на рисунке 25 для $\nu=1$ очень похоже на $u(t)$ из [99], для $\nu=5$, наше u для $\nu=2$ — на $u(t)$ из [99] при $\nu=10$ и т. д. Основная причина, видимо, в слишком осторожной тактике назначения S в расчетах [99] — излишне жестко требование иметь на каждой итерации $|x^1 (T) - 0,2| \leq \varepsilon$ ($\varepsilon \approx 0,001$), причем, в отличие от наших расчетов, погашение невязки в условии $x^1 (T) - 0,2 = 0$ не включается в задачу определения $\delta u(t)$, и приходится следить, чтобы не происходило накопления погрешности. Автор в своих расчетах придерживался более либеральных требований, считая необходимым

*) На 1—6 и 12-й итерациях.

удовлетворить условиям с нужной точностью лишь в конце расчета. А в процессе поиска допускались какие-то отклонения. Это позволяло использовать большой шаг S . Заметим, однако, что использование большого S в методике [99] может привести к некоторым неприятностям. Дело в том, что замена (10) порождает

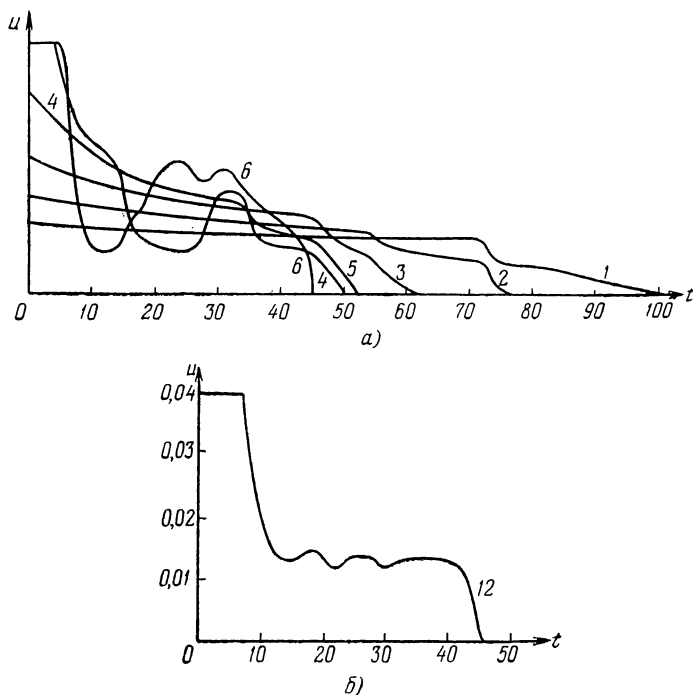


Рис. 25.

«прилипание» управления к границе. Этот факт объясняется в § 18, он отмечен и в [99]. Дефект замены (10) может и не иметь неприятных последствий, если процесс оптимизации протекает монотонно: если управление $u(t)$ где-то вышло на границу интервала $[0, U^+]$, то в дальнейшем не будет необходимости на каком-то участке снова перейти к ситуации $0 < u < U^+$. В наших расчетах с крупным шагом S эволюция $u(t)$ в целом носит монотонный характер, однако встречаются и нарушения монотонности (см. рис. 25). Может быть, это еще одна причина, заставляющая в методике [99] придерживаться сравнительно малого шага S . В этой задаче была построена область достижимости. Для этого

находились оптимальные значения $x^2(T)$ при разных значениях $x^1(T)$. На рис. 26 изображена величина $\max x^2(T)$ как функция от $x^1(T)$ *). Она отделяет достижимые точки $\{x^1(T), x^2(T)\}$ от недостижимых. Обратите внимание на вогнутость области достижимости.

§ 30. Задача о плоском движении тела переменной массы

Траектория управляемой системы определяется уравнениями

$$\begin{aligned} \dot{x}^1 &= -u_1, \\ \dot{x}^2 &= x^4 \cos x^5, \\ \dot{x}^3 &= x^4 \sin x^5, \\ \dot{x}^4 &= g \left[\frac{Vu_1 - C_1(x^4, x^3, u_2)}{x^1} - \sin x^5 \right], \\ \dot{x}^5 &= \frac{g}{x^4} \left[\frac{Vu_1 - C_2(x^4, x^3)}{x^1} u_2 - \cos x^5 \right], \\ 0 &\leq t \leq T. \end{aligned} \quad (1)$$

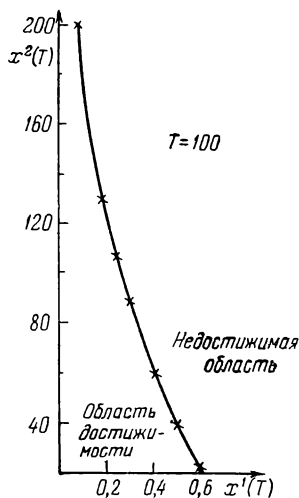


Рис. 26.

Здесь x^1 — масса, x^2, x^3 — координаты, x^4 — абсолютная величина скорости, x^5 — угол наклона траектории. Уравнения описывают движение реактивного самолета (или ракеты). Управлениями являются: $u_1(t)$ — секундный расход массы, $u_2(t)$ — угол атаки. $C_1(x^4, x^3, u_2)$, $C_2(x^4, x^3)$ — некоторые заданные функции, характеризующие аэродинамические свойства объекта и изменение плотности с высотой (x^3). V, g — заданные постоянные. Система (1) дополняется условиями:

$$\Gamma(x) = 0: \quad x(0) - X_0 = 0 \quad (\text{данные Копи}), \quad (2)$$

$$u(t) \in U: \quad 0 \leq u_1(t) \leq U_1^+; \quad |u_2(t)| \leq U_2. \quad (3)$$

Качество управления характеризуется следующими функционалами (время управления T не фиксировано):

$$\begin{aligned} F_0[u(\cdot), T] &\equiv x^1(T), \\ F_1[u(\cdot), T] &\equiv x^2(T) - X_2, \\ F_2[u(\cdot), T] &\equiv x^3(T) - X_3 \end{aligned} \quad (4)$$

(X_2, X_3 — заданы).

*) Значения, отмеченные знаком «X», получены решением вариационных задач.

Здесь будут в общих чертах приведены результаты решения ряда вариационных задач (1)–(3). Они решались методом последовательной линеаризации (§§ 19–21) еще в 1962–1963 гг., когда технология метода только начинала складываться и проходила проверку. Поэтому мы остановимся лишь на некоторых деталях. Прежде всего заметим, что функции C_1 и C_2 были заданы достаточно сложными выражениями, являющимися суперпозицией вспомогательных функций, в том числе и заданных таблично. Поэтому при решении сопряженной системы $\dot{\psi} = -f_x^* \psi$ вычисление матрицы $f_{xx}(x(t), u(t))$, f_{xu} осуществлялось численным дифференцированием, т. е. составлялась программа только для вычисления правых частей $f(x, u)$ (1). Здесь следует принять некоторые предосторожности, связанные с использованием функций, заданных таблично. Обычно подобные таблицы содержат небольшое число значений для набора узлов в области изменения независимого аргумента, а между ними функция интерполируется линейно, так как применение более точных методов интерполяции не оправдано ввиду неточности самих табличных значений (как правило, таблицами задаются функциональные зависимости экспериментального характера). Однако для наших целей нужны дифференцируемые функции $f(x, u)$, поэтому следует предпочесть гладкие методы восполнения таблично заданной функции (например, с помощью сплайнов).

Успешное решение задач для системы (1)–(3) во многом связано с использованием естественных единиц измерения для различных функционалов и компонент управления. Это были первые (в практике автора) задачи с двумя управляющими функциями, имеющими существенно разные физические размерности. Попытки решения их в естественных физических единицах измерения сразу же привели к трудностям, осмысливание причин первых неудач и выявило важность согласования единиц измерения разнородных объектов. Для системы (1) решались задачи, имеющие прямое прикладное значение. Однако здесь мы приведем примеры решения задач, имеющих иллюстративный характер. Выбор в пользу именно этих задач, кроме всего прочего, определяется тем, что они оказались более интересными, их решения не так монотонны, как решения реальных. Отличаются же они от содержательных задач только значениями X_2 , X_3 в выражениях (4).

Первая задача. Найти $\max F_0[u(\cdot), T]$ при условиях $F_1 = F_2[u(\cdot), T] = 0$ ($X_2 = 2,00$; $X_3 = 0,02$; единицы условные). Другими словами, ставится задача минимизации расхода топлива (максимум конечного веса) при условии попадания в заданную точку. Процесс решения задачи показан в табл. 1. Значение $x^3(T)$ не представлено в таблице, так как условие $x^3(T) = X_3$ использовалось в качестве признака окончания интегрирования (т. е. определения T). Примерно таким же образом решались и

Т а б л и ц а 1

№ итерации	$x^1(T)/X(0)$	$x^2(T)$	T	№ итерации	$x^1(T)/X(0)$	$x^2(T)$	T
2	0,4849	1,898	175	16	0,6571	1,957	295
4	0,5400	1,964	192	18	0,6631	1,964	304
6	0,5906	1,900	205	20	0,6678	1,930	306
8	0,6086	1,975	227	22	0,6672	1,951	309
10	0,6256	1,994	254	24	0,6650	2,000	322
12	0,6322	2,004	263	26	0,6650	2,010	315
14	0,6461	2,000	275	28	0,6683	1,985	313

остальные задачи. Таблица показывает, что начиная с 20-й итерации значение F_0 более или менее стабилизировалось. Следует ли из этого оптимальность найденной траектории (приближенная, разумеется), или просто метод стал неэффективным, и медленную сходимость, замаскированную к тому же легкой «болтанкой»

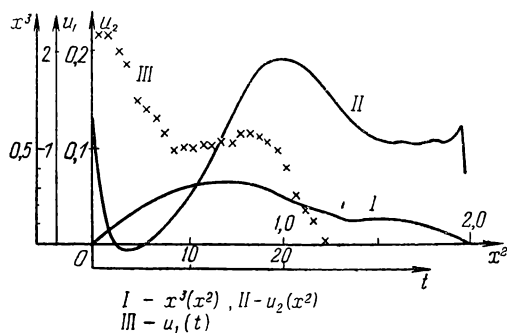


Рис. 27.

значений $x^1(T)$ и $x^2(T)$, мы воспринимаем, как достижение практического экстремума? Подобный вопрос всегда стоит перед вычислителем, занимающимся решением практических задач. Некоторую информацию доставляет нам анализ конуса вариаций K_u и конуса смещений K_F (см. § 5). Обратимся прежде всего к рис. 27, на котором показаны «траектория» $x^3(x^2)$ и управления $u_1(t)$, $u_2(x^2)$. Всюду $u_1(t) < U_1^+$ и $u_1(t) > 0$ при $t < 24$, $u_1(t) = 0$ при $t > 24$. Кроме того, $|u_2(t)| < U_2$ при всех t . Поэтому структура K_u следующая:

$$\delta u(t) \begin{cases} > 0 & \text{при } t > 24, \\ \text{произвольна} & \text{при } t < 24, \end{cases}$$

$\delta u_2(t)$ — произвольна, δT — произвольна.

Конус K_u отображается в конус смещений K_F , состоящий из трехмерных векторов $\delta F[\delta u(\cdot), \delta T] = \{\delta F_0, \delta F_1, \delta F_2\}$,

$$\delta F_i[\delta u(\cdot), \delta T] = \int_0^T [w_i^1(t)\delta u_1(t) + w_i^2(t)\delta u_2(t)] dt + a_i\delta T, \\ i=0, 1, 2,$$

причем, как нетрудно заметить, $w_0^1(t) \equiv 1$, $w_0^2(t) = 0$ (изменение угла атаки не влияет на конечную массу). Геометрическую структуру конуса K_F легко представить по рис. 28. На нем значками «•»

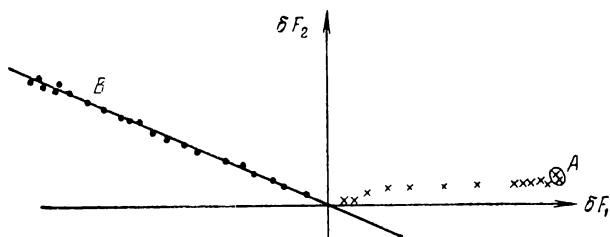


Рис. 28.

нанесены векторы $\{0, w_1^2(t), w_2^2(t)\}$ (для разных значений t). Они с хорошей точностью располагаются на прямой B в плоскости $\delta F_0 = 0$. Так как $\delta u_2(t)$ — произвольна, то

$$K'_F: \quad \delta F = \int_0^T w^2(t)\delta u_2(t) dt$$

есть прямая B (это лишь часть конуса K_F). На том же рисунке значками « \times » нанесены векторы $\{1, w_1^1(t), w_2^1(t)\}$; так как $w_0^1(t) \equiv 1$, то « \times » нужно представлять себе лежащими в плоскости $\delta F_0 = 1$. Эти векторы разбиваются на две группы: одна, соответствующая интервалу $0 < t < 24$, представлена практически одной точкой A . Так как на $[0; 24]$ $\delta u_1(t)$ произвольна, то более полное приближение к K_F

$$K''_F: \quad \delta F = \int_0^T w^2(t)\delta u_2(t) dt + \int_0^{24} w^1(t)\delta u_1(t) dt$$

есть плоскость Π , натянутая на прямую B и точку A в плоскости $\delta F_0 = 1$. Векторы $\{1, w_1^1(t), w_2^1(t)\}$, соответствующие интервалу $(24, t)$ — это остальные « \times », также расположенные в плоскости

$\delta F_0 = 1$, однако в конструкции $\int_{24}^T w^1(t) \delta u(t) dt$ они могут быть взяты только с весом $\delta u_1(t) \geq 0$. Учитывая и их, мы превратим K_F'' в конус

$$K_F'': \quad \delta F = \int_0^T w'(t) \delta u(t) dt + \int_0^T w^2(t) \delta u_2(t) dt.$$

Легко сообразить, что K_F'' есть полупространство, ограниченное плоскостью Π и лежащее ниже ее, т. е. не содержащее направления $\{1; 0; 0\}$, а это и есть принцип максимума (ведь мы решаем задачу на $\max F_0$). Разумеется, следовало бы еще проанализировать

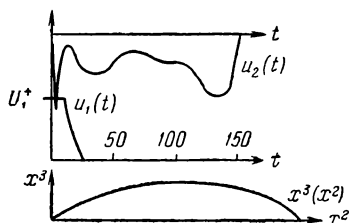


Рис. 29.

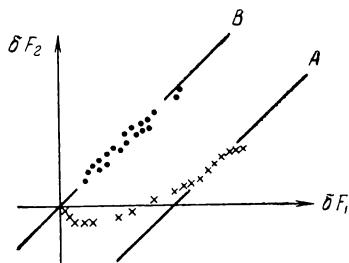


Рис. 30.

расширение K_F'' до K_F за счет включения векторов $a \delta T$ (δT — произвольно). К сожалению, данные о векторе a не сохранились, и это мешает нам проиллюстрировать выполнение условия трансверсальности: вектор a должен, разумеется, лежать в плоскости Π . Заметим, что этот расчет, как и все остальные, начинался с достаточно грубого приближения. Обычно $u_2(t) \equiv 0$, $u_1(t) = C$ при $t < t'$, $u_1(t) = 0$ при $t > t'$, C и t' задавались величинами разумными, но не более того.

Вторая задача. Она была решена с целью проконтролировать решение первой и была к ней «двойственной»: найти $\max_{u, T} x^2(T)$ при условиях $F_0 = 0,6680$, $F_2 = 0$. Ее решение должно совпадать с решением первой, и такое совпадение было получено достаточно надежно. Подробно комментировать задачу не будем.

Третья задача. От первой она отличалась только тем, что T было исключено из управления и фиксировано: $T = 150$. Кроме того, в первой задаче $x^4(0) = 0,4$, в третьей — $x^4(0) = 0,75$. Решение представлено на рис. 29 теми же функциями: $x^3(x^2)$,

$u_1(t), u_2(t)$. По-прежнему всюду $|u_2(t)| < U_2$, а

$$u_1(t) = \begin{cases} = U_1^+ & \text{при } t \in (0, t'), \\ \in (0, U_1^+) & \text{при } t \in (t', t''), \\ = 0 & \text{при } t \in (t'', T). \end{cases}$$

Конус K_u теперь имеет следующую структуру

$$\delta u_1(t) \begin{cases} \leq 0 & \text{на } (0, t'), \\ \text{произв.} & \text{на } (t', t''), \delta u_2 \text{ — произвольна,} \\ \geq 0 & \text{на } (t'', T). \end{cases}$$

Рис. 30, построенный по тому же принципу, что и рис. 28, позволяет представить строение конуса K_F . Векторы $w^2(t)$, лежащие в плоскости $\delta F_0 = 0$, группируются около прямой B , и конус

$$K'_F: \quad \delta F = \int_0^T w^2(t) \delta u(t) dt \quad (\delta u_2(t) \text{ — произвольна})$$

есть прямая B . Векторы $w^1(t)$ представлены на рис. 30 точками $\{w_1^1, w_2^1\}$, их следует считать лежащими в плоскости $\delta F_0 = 1$. Они теперь разбиты на три группы. Те, которые соответствуют интервалу (t', t'') , где разрешена произвольная вариация $\delta u_1(t)$, расположились на прямой A (в плоскости $\delta F_0 = 1$). Параллельность A и B хорошо видна на рисунке. Конус

$$K''_F: \quad \delta F = \int_0^T w^2(t) \delta u_2(t) dt + \int_{t'}^{t''} w^1(t) \delta u_1(t) dt \\ (\delta u_2(t), \delta u_1(t) \text{ — произвольны})$$

есть плоскость Π , проходящая через A и B . Наконец, включая в конструкцию K_F группу $w^1(t)$, $t \in (0, t') \cup (t'', T)$, расширяем Π до полупространства, лежащего ниже Π . Таким образом и здесь проверено выполнение принципа максимума. Разумеется, речь идет о приближенном выполнении, и естественно возникает вопрос о том, можно ли отклонения, например, точек «•» на рис. 30 от прямой B считать малыми и пренебречь ими. Некоторые дополнительные сведения дает сравнение чертежей типа 28, 30 с аналогичными, но построенными для траекторий, найденных в самом начале процесса решения задачи. Такое сравнение помогает составить представление о том, что есть малая величина.

Данные о конусах K_F на исходных траекториях не сохранились, и читателю остается поверить автору, что разница между K_F в исходной и численно оптимальной ситуации была такой же, какова она на аналогичных рисунках в § 31—33.

Разумеется, строгий математик скажет, что подобные рисунки ничего не доказывают, и что он готов построить пример задачи, в которой будет выполнен принцип максимума с той точностью, которая получилась в наших расчетах, и тем не менее траектория не оптимальна, и очень грубо не оптимальна. Однако основным назначением численных методов является решение задач, возникших в приложениях. Поэтому анализ, подобный проделанному выше, хотя и не обладает силой и бесспорностью доказательства, оказывается полезным и позволяет избежать грубых просчетов.

Ч е т в е р т а я з а д а ч а. По сравнению с первой она была осложнена введением дополнительного условия $x^5(T) = X_5$. Так как программа была рассчитана только на два дополнительных условия, решение удалось провести с помощью несложного приема: от системы (1) вида $\dot{x} = f(x, u)$ перешли к системе, в которой координата x^2 стала независимым переменным

$$\frac{dx^1}{dx^2} = \frac{f^1}{f^2}; \quad \frac{dx^3}{dx^2} = \frac{f^3}{f^1}; \quad \frac{dx^4}{dx^2} = \frac{f^4}{f^2}; \quad \frac{dx^5}{dx^2} = \frac{f^5}{f^2}; \quad 0 \leq x^2 \leq X_2.$$

Эта замена оказалась возможной потому, что в искомом решении функция $x^2(t)$ должна была быть заведомо монотонной. Независимость $f(x, u)$ от t тоже была полезной, но не очень существенной: в крайнем случае можно было бы добавить пятое уравнение $dt/dx^2 = 1/f^2$ (ведь программа все равно была рассчитана на пять уравнений). Существенным, конечно, было то, что время процесса T не было фиксированным. Мы не будем подробно комментировать решение задачи. Оно изображено на рис. 3 в [87], а на рис. 6 (там же) — конус для этой задачи. И здесь принцип максимума оказался выполненным (с той же степенью точности). Приведенные здесь расчеты выполнены в 1963 г. и опубликованы в [83], [89].

§ 31. Оптимизация химического реактора

В этом параграфе будет рассмотрена и численно решена довольно простая задача, связанная с оптимизацией некоторого химического аппарата. Фазовое пространство — трехмерно: $x = \{x^1, x^2, x^3\}$. Система дифференциальных уравнений

$$\begin{aligned} \frac{dx^1}{dt} &= -[k_1(u) + k_2(u) + k_3(u)]x^1, \\ \frac{dx^2}{dt} &= k_1(u)x^1 - k_4(u)x^2, \\ \frac{dx^3}{dt} &= k_4(u)x^2 - k_5(u)x^3, \quad 0 \leq t \leq T, \end{aligned} \quad (1)$$

описывает реакции, протекающие в смеси трех веществ, $x^i(t)$ — их концентрации. Интенсивности реакций зависят от температуры

$u(t)$, играющей в данной задаче роль управления. Первое вещество, концентрация которого $x^1(t)$, есть сырье, второе — промежуточный продукт, третье — окончательный. Начальные данные

$$\Gamma(x)=0: \quad x^1(0)=1=0; \quad x^2(0)=0; \quad x^3(0)=0. \quad (2)$$

Ограничение на управление $u(t) \in U$:

$$0 \leq u \leq 823, \quad (3)$$

Будут рассмотрены два функционала:

$$F_0[u(\cdot), T] \equiv -x^3(T); \quad F_1[u(\cdot), T] \equiv x^2(T) - X_2. \quad (4)$$

Время реакции T не фиксировано, а ищется одновременно с $u(\cdot)$. Что касается функций $k_i(u)$, то они имеют характерный для химической кинетики вид

$$k_i(u) = C_i \exp\left(\frac{E_i}{R} \left(\frac{1}{658} - \frac{1}{u}\right)\right). \quad (5)$$

Значения постоянных взяты из книги [70], где эта задача решается различными методами. Мы воспроизведем здесь один из применявшихся в [70] методов (наиболее успешный), сравним с нашими результатами, и обсудим причины некоторых трудностей численного решения в [70]:

$$C_1=1,02; \quad C_2=0,93; \quad C_3=0,386; \quad C_4=3,28; \quad C_5=0,084; \\ E_1=16\,000; \quad E_2=14\,000; \quad E_3=15\,000; \quad E_4=10\,000; \quad E_5=15\,000; \\ R=1,9865.$$

Основным источником вычислительных затруднений является экспоненциальная зависимость коэффициентов k_i от u .

В [70] решалась простейшая неклассическая задача: найти $u(\cdot)$, на котором достигается

$$\min_{u(\cdot)} F_0[u(\cdot)] \quad (\text{т. е. } \max x^3(T)). \quad (6)$$

Величина T была фиксирована, однако затем решалась серия задач с разными T , что позволило найти и оптимальное T . В [70] использовался метод проекции градиента: на данной траектории $\{u(\cdot), x(\cdot)\}$ вычислялась производная функционала $F_0[u(\cdot)]$

$$w_0(t) = \frac{\partial F_0[u(\cdot)]}{\partial u(\cdot)},$$

и в качестве следующего управления бралась функция

$$P_U[u(t) + sw_0(t)],$$

где P_U — операция проектирования на множество U , реализующаяся очевидным способом:

$$P_U[v(t)] \equiv \begin{cases} 823, & \text{если } v(t) > 823, \\ v(t), & \text{если } v(t) \leq 823. \end{cases}$$

Скалярный параметр s — шаг процесса, подбираемый экспериментально. В качестве начального управления бралась функция $u(t)=673$. В табл. 1 приведены результаты серии расчетов с разными s при $T=1$. Заметим, что наибольшее значение выхода окончательного продукта при $T=1$, которое было получено в расчетах [70], есть 0,435.

Т а б л и ц а 1 [70]

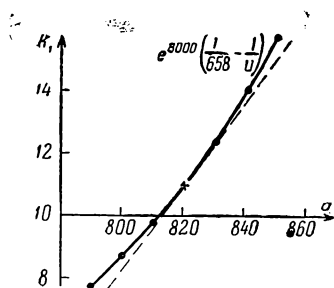
Номер итерации	$s = 0,1$	$s = 0,2$	Номер итерации	$s = 0,45$	ν	$s = 0,75$
0	0,357	0,357	0	0,357	0	0,357
1	0,357	0,357	1	0,356	1	0,358
10	0,359	0,361	10	0,367	10	0,374
50	0,367	0,389	23	0,409	16	0,417
100	0,390	0,425	26	∞	18	∞
110		0,435				

Из таблицы видно, что расчет с $s=0,2$ дал хороший результат, однако процесс поиска был слишком медленным. Причина этого ясна — малая величина шага s . Попытки поиска с большим шагом s ($s=0,5$, $s=0,75$) вначале были явно эффективнее, однако до конца не были доведены из-за выхода в физически бессмысленную область $u < 0$, $x^i > 1$. Отчасти неудача этих расчетов была связана с тем, что не было поставлено легко учитываемое условие типа $u \geq 0$, однако основная причина — это расчет с постоянным шагом s . Нет никаких сомнений в том, что расчет с $s=0,75$ был бы благополучно доведен до конца, если бы после 16-й итерации произошло соответствующее уменьшение шага s (например, до величины 0,2 или 0,1). Решение задачи (1)–(3), (6) было повторено с использованием технологии § 18. На этом простом примере нам будет удобно пояснить те ориентировочные оценки, которые обычно предшествуют численному решению задачи. Первый вопрос, который здесь возникает, это вопрос о том, какие величины вариации u допустимы (с точки зрения точности линейного приближения) и какие вариации желательны для достаточно быстрого решения задачи. Поскольку $u(t)$ ограничено значением 823 °K, а температуры ниже 473 °K технологически невыгодны (это довольно элементарное содержательное свойство системы, которое легко получить, оценив k_i (473°)), то можно предположить, что максимальное «расстояние» от исходного $u(t)$ до оптимального $u^*(t)$ есть

$$\max_t |u(t) - u^*(t)| \approx 150 - 200.$$

Желая решить задачу за 10–20 шагов, мы должны работать с вариациями управления $|\delta u| \approx 20$. Теперь посмотрим, допустимы ли

такие δu с точки зрения точности линеаризации задачи? Основное подозрительное место — это экспоненциальная зависимость скорости реакции от u , причем нелинейность этой зависимости особенно сильно проявляется при высоких температурах. Рассмотрим функцию $k_1(u) \approx \exp\left(8000\left(\frac{1}{658} - \frac{1}{u}\right)\right)$. На рис. 31 приведен график $k_1(u)$ в окрестности точки $u=823$, из которого видно, что эта функция может быть с достаточной точностью заменена касательной на интервале, например, $[820-30, 820+30]$. Ясно, что при меньших температурах функция $k_1(u)$ линеаризуется с достаточной точностью на больших интервалах. Это простое исследование показывает, что мы не должны встретить серьезных вычислительных затруднений, если ограничим вариацию управления



$|\delta u(t)|$ величиной $\sim 10-20$. Разумеется, этот вывод не вполне строг, но с него можно начать, предполагая более точное представление о задаче получить в ходе ее решения. Мы будем решать задачу: найти

$$\min_{u(\cdot), T} F_0[u(\cdot), T]$$

при условии

$$F_1[u(\cdot), T] = 0,$$

Рис. 31

т. е. кроме максимизации $x^3(T)$ следует еще обеспечить заданный числом X_2 выход промежуточного вещества. Решение будет осуществляться методом проекции градиента. Напомним в общих чертах суть дела: интервал $[0, T]$ разбивается на N (в расчетах $N=100$) равных частей (t_n, t_{n+1}) , $n=0, 1, \dots, N-1$, и ищется кусочно-постоянное управление $u_{n+1/2}$, одновременно ищется и T . При заданных $\{u_{n+1/2}, T\}$ система (1) интегрируется с начальными данными (2), и находятся значения функционалов $F_0[u(\cdot), T]$, $F_1[u(\cdot), T]$. Затем дважды интегрируется сопряженная система с начальными данными при $t=T$: $\psi(T) = \{0, 0, -1\}$, $\psi(T) = \{0, 1, 0\}$, что позволяет найти функциональные производные

$$w_i(t) = \frac{\partial F_i[u(\cdot), T]}{\partial u(t)}, \quad a_i = \frac{\partial F_i[u(\cdot), T]}{\partial T}, \quad i=0, 1.$$

Далее, решение задачи квадратичного программирования определяет вариацию управления, т. е. набор чисел $\{\delta u_{n+1/2}\}$, δT .

На функцию $\delta u(t)$ было наложено ограничение $s^-(t) \leq \delta u(t) \leq s^+(t)$, где

$$s^+(t) = \min\{30, 823 - u(t)\},$$

$$s^-(t) = \max\{-30, u_{\min} - u(t)\}.$$

На вариацию δT было наложено ограничение $|\delta T| \leq 0,07$ T. Наконец, отметим еще одно место, требующее внимания при организации вычислений. При высоких температурах ($u \sim 823$ °K) некоторые реакции идут очень энергично. Это означает, что в системе уравнений (1) первое, например, уравнение принимает вид

$$\dot{x}^1 \simeq -20x^1.$$

Достаточно точное интегрирование такого уравнения методом второго, например, порядка точности (в наших расчетах использовался метод Эйлера с пересчетом) требует шага $\tau \approx 0,001$. Поэтому шаг интегрирования дифференциальных уравнений (как прямого (1), так и сопряженных) не совпадал с шагом сетки для u (его величина $\Delta t \approx 0,01$), а был в 10 раз меньшим. Что касается числа S , то вначале оно задавалось величиной 100, а затем в процессе решения изменялось так, чтобы среднее значение $|\delta u(t)|$ было порядка 20.

Решение задачи (с дополнительным условием $x^2(T) = X_2$) было проведено в нескольких вариантах.

В а р и а н т 1. $X_2 = 0,0437$, кроме того, число $S = 100$ не менялось в процессе поиска. Ход решения показан в табл. 2 следующими величинами: ν — номер итерации, T , $x^3(T)$, значение $x^3(T)$, предсказанное на предыдущем этапе по формулам линейной теории возмущений, значение $x^2(T)$ и предсказанное значение $x^2(T)$, среднее значение вариации $|\delta u(t)|$.

Т а б л и ц а 2

ν	T	$x^3(T)$ фактич.	$x^3(T)$ предск.	$x_2(T)$ фактич.	$x^2(T)$ предск.	δu среднее
0	1,000	0,35679	—	0,043697		
1	0,930	0,36271	0,3615	0,0434	0,0437	6,1
2	0,865	0,37142	0,3690	0,0429	0,0437	8,1
3	0,804	0,38214	0,3794	0,0429	0,0437	9,9
4	0,748	0,39475	0,3927	0,0431	0,0437	11,0
5	0,696	0,40856	0,4078	0,0435	0,0437	12,0
6	0,672	0,41282	0,4127	0,0436	0,0437	5,4
7	0,672	0,41391	0,4137	0,0436	0,0437	3,8
8	0,675	0,41463	0,4145	0,0436	0,0437	3,5
10	0,684	0,41550	0,4154	0,0436	0,0437	3,1
12	0,695	0,41614	0,4161	0,0437	0,0437	2,7
14	0,707	0,41642	0,4164	0,0437	0,0437	2,5
15	0,713	0,41659	0,4166	0,0437	0,0437	2,1

В этом расчете начальная функция $u(t) = 673$; резкое сокращение $|\delta u(t)|$ после пятой итерации связано с тем, что на части интервала $[0, T]$ управление достигло верхней границы 823.

В а р и а н т 2. Единственное отличие его от первого состоит в том, что использовался алгоритм пересчета S с тем, чтобы обеспечить заданную среднюю величину вариации $|\delta u| \sim 20$. Результаты представлены в табл. 3 величинами: ν , $x^3(T)$, предсказанное $x^3(T)$, S и среднее значение $|\delta u|$. Величина $x^2(T)$ ведет себя примерно так же, как в табл. 2. Пересчет S ускорил решение по

Т а б л и ц а 3

ν	T	$x^3(T)$ фактич.	$x^3(T)$ предск.	S	$ \delta u $ среднее
0	1,000	0,35679	—	100	—
1	0,930	0,36271	0,3615	330	6,1
2	0,865	0,37509	0,3702	420	16
3	0,804	0,38949	0,3847	360	23
4	0,748	0,40423	0,4026	260	27
5	0,696	0,41584	0,4160	230	23
6	0,661	0,41675	0,4169	1100	4
7	0,675	0,41724	0,4170	3200	7
8	0,702	0,41732	0,4173	7700	8,4

меньшей мере вдвое. Естественно, возникает вопрос, в какой мере можно считать процесс поиска оптимального управления законченным. Характер изменения $x^3(T)$ до некоторой степени свидетельствует об оптимальности: хотя вариация управления не очень

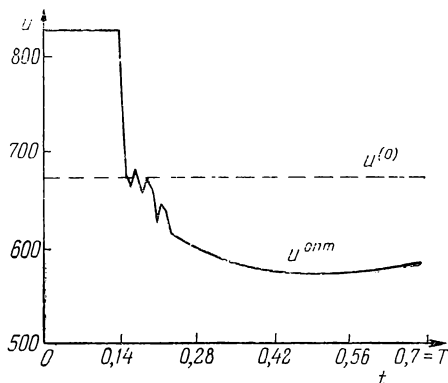


Рис. 32.

мала ($|\delta u| \sim 8$), $x^3(T)$ практически не меняется. Более убедительным является анализ конуса достижимости, построенный для траектории на восьмой итерации. Этот конус изображен на рис. 33, однако для понимания того, что изображено, необходимы некоторые пояснения.

Прежде всего опишем конус допустимых по условию $u(t) \in U$ вариаций управления. Полученное на восьмой итерации управление $u(t)$ на отрезке $[0; 0,14]$ достигло верхнего допустимого значения 823; на $(0,14; T)$ $u(t)$ находится строго внутри U . Поэтому

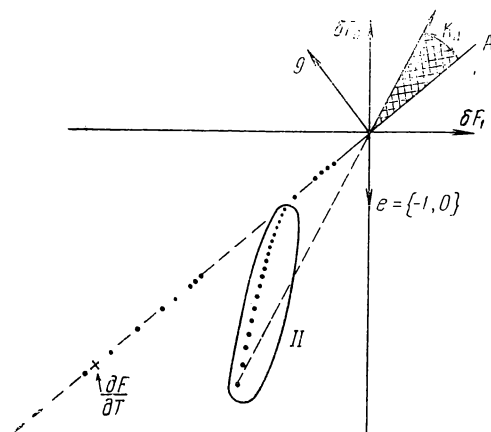


Рис. 33.

конус K_u есть множество функций $\delta u(t)$, удовлетворяющих условиям

$$\begin{aligned} \delta u(t) &\leq 0 \text{ при } t \in (0; 0,14), \\ \delta u(t) &\text{ произвольно при } t \in (0,14; T). \end{aligned}$$

Напомним, что основной информацией, по которой строится вариация управления, является совокупность векторов

$$h_{n+1/2} = \begin{Bmatrix} \frac{\partial F_0}{\partial u_{n+1/2}} \\ \frac{\partial F_1}{\partial u_{n+1/2}} \end{Bmatrix}, \quad h_{N+1/2} = \begin{Bmatrix} \frac{\partial F_0}{\partial T} \\ \frac{\partial F_1}{\partial T} \end{Bmatrix}.$$

На рис. 33 нанесены точки $h_{n+1/2}$. Они четко разделяются на две группы: первая группа точек $h_{n+1/2}$, с хорошей точностью укладывающихся на прямую A , соответствует тем интервалам сетки, которые попадают на интервал $(0,14; T)$. Соответствующие $\delta u_{n+1/2}$ — произвольны, поэтому множество смещений δF , вычисляемых по формуле

$$\delta F_I = \sum_I \delta u_{n+1/2} h_{n+1/2}$$

(суммирование только по индексам первой группы) с произвольными $\delta u_{n+1/2}$, образует, линейное пространство, вырождающееся

в прямую A (разумеется, если несколько идеализировать картину, считая $h_{n+1/2}$ лежащими на A). Точки $h_{n+1/2}$ второй группы лежат вне A ; они соответствуют тем интервалам сетки, которые попадают на интервал $(0; 0,14)$ и порождают множество смещений δF , определяемых формулой

$$\delta F_{II} = \sum_{II} \delta u_{n+1/2} h_{n+1/2}, \quad \delta u_{n+1/2} \leq 0.$$

Множество всех таких δF заполняет изображенный на рис. 33 конус K_{II} . Всевозможные смещения

$$\delta F = \sum_I \delta u_{n+1/2} h_{n+1/2} + \sum_{II} \delta u_{n+1/2} h_{n+1/2},$$

заполняют полуплоскость выше прямой A . Однако имеется еще вектор $h_{N+1/2} = \partial F / \partial T$, причем δT может иметь произвольный знак. Этот вектор также попадает на A и не меняет картины. Итак, конус возможных смещений K_F , образованный всеми допустимыми возмущениями $\{\delta u_{n+1/2}, \delta T\}$, есть полуплоскость, не содержащая «запрещенного» направления $e = \{-1, 0\}$. Это означает, что найденная траектория удовлетворяет принципу максимума, фигурирующий в нем вектор g (см. § 5) есть нормаль к A , а тот факт, что $\partial F / \partial T \in A$, свидетельствует о выполнении условия трансверсальности.

Теперь мы можем вернуться к сравнению наших расчетов с расчетами в [70] (см. табл. 1). Видно, что решение задачи в [70] было неоправданно трудоемким; оно потребовало почти в 15 раз большего числа итераций. Единственной причиной был неудачный способ регулирования величины вариации $\delta u(t)$ параметром s , принятый в [70]; точнее, неудачным является выбор единого s на весь процесс. В начале поиска при $u \approx 673$ влияние изменения температуры на $x^3(T)$ очень мало, и величина $s\omega_0 = \delta u$ при $s=0,2$ приводит к среднему изменению u на 0,2. Поэтому начальная стадия поиска проходит неоправданно медленно. Постепенно температура повышается, увеличивается эффективность управления (т. е. $\omega_0(t)$), и вариация $\delta u = s\omega_0(t)$ становится более разумной. Попытка же ускорить начало процесса, взяв $s=0,75$, привела к тому, что высокие температуры были достигнуты достаточно быстро (на 16-й итерации u в расчетах [70] достигло при $t \approx 0$ величины 818), однако теперь вариация $\delta u(t) = s\omega_0(t)$ становится неоправданно большой с точки зрения применимости линейной теории возмущений и приводит к нелепым результатам. Разумеется, исправить метод [70] было бы очень просто, для нас же важен следующий вывод: величина вариации $\delta u(t)$ имеет простой содержательный смысл, и указать естественное ограничение для нее не очень сложно (используя простые средства, аналогичные рис. 1). Что касается величины s (как в методе [70], так и в нашем методе), то ее «естественное» значение заранее не очевидно. Однако

можно (и не очень сложно) построить алгоритмы подбора в процессе решения значения s , обеспечивающего близкую к запланированной величину вариации $|\delta u(t)|$. В расчетах подбор S делался так: после очередной вариации управления вычислялась средняя величина $\Delta_{\text{ср}} = \frac{1}{T} \int_0^T |\delta u| dt$ и пересчитывалась величина S по формуле $S := S\Delta/\Delta_{\text{ср}}$, где Δ — запланированная средняя величина δu (см. табл. 3).

§ 32. Оптимизация производственного цикла

Математическая модель производства некоторого вещества приводит к системе уравнений с управлением

$$dx/dt = Ax + uBx; \quad 0 \leq t \leq T. \quad (1)$$

Здесь $x(t) = \{x^1, x^2, x^3\}$ — вектор-функция, компоненты которой описывают изменение концентраций трех участвующих в реакции веществ: x^1 — концентрация исходного вещества (сырья), x^2, x^3 — концентрации промежуточного и окончательного продуктов. Система (1) описывает протекающие в смеси реакции; при этом uBx — это скорости реакций, протекающих под воздействием некоторого фактора, поддающегося регулированию и выступающего в математической модели в качестве управления. На значения $u(t)$ наложено физическое ограничение

$$u(t) \in U; \quad 0 \leq u(t) \leq 1. \quad (2)$$

Время, в течение которого в смеси веществ происходят взаимные превращения, будем считать фиксированным и равным T . В момент времени T из смеси изымается весь окончательный продукт (x^3), а оставшаяся смесь x^1 и x^2 , в которой количество «сырья» x^1 по-прежнему до первоначальной концентрации, вновь подвергается такой же обработке. Рассматривается задача оптимизации подобного цикла. Таким образом, краевые условия в задаче имеют вид

$$Gx = 0; \quad x^1(0) = 1; \quad x^3(0) = 0; \quad x^2(0) = x^2(T). \quad (3)$$

Задача состоит в минимизации функционала

$$F_0[u(\cdot)] \equiv 1 - x^1(T) \quad (4)$$

(расход сырья) при заданном «уровне производства» D

$$F_1[u(\cdot)] \equiv x^3(T) - D \geq 0. \quad (5)$$

Здесь представлена несколько упрощенная схема задачи. В действительности был еще некоторый интервал времени $[T, T + \Delta T]$, на котором происходил отбор продукта при $u(t) = 0$, причем этот процесс сопровождался изменениями концентраций веществ.

(3 \rightarrow 3)-матрицы A и B — постоянны; точный вид их и физический смысл задачи для дальнейшего не очень важен, и мы его разяснять не будем. Можно было бы решать задачу в той форме, в какой она представлена выше. Тогда определение фазовой траектории $x(t)$ и вычисление функционалов F_0, F_1 при заданном управлении $u(t)$ требуют решения краевой задачи (1), (3) с условиями типа условий периодичности. Такой же характер имели бы и краевые задачи для сопряженных переменных $\psi(t)$, с помощью которых вычисляются функциональные производные F_0 и F_1 . Для того чтобы иметь дело только с задачами Коши, был введен управляющий параметр α , и задача приобрела следующую стандартную форму:

1. Система уравнений

$$\dot{x} = f; \quad \dot{x} = Ax + uBx.$$

2.

$$\Gamma x = 0; \quad x^1(0) = 1; \quad x^2(0) = \alpha; \quad x^3(0) = 0.$$

3.

$$\min_{u, \alpha} F_0[u(\cdot), \alpha], \quad F_0[u(\cdot), \alpha] \equiv 1 - x^1(T).$$

4. Дополнительные условия:

$$F_1[u(\cdot), \alpha] \equiv x^3(T) - D \geq 0, \\ F_2[u(\cdot), \alpha] \equiv x^2(T) - \alpha = 0.$$

Теперь для вычисления функционалов F_0, F_1, F_2 по заданному управлению $\{u(\cdot), \alpha\}$ достаточно проинтегрировать задачу Коши. Вычисление производных

$$\frac{\partial F_i[u(\cdot), \alpha]}{\partial u(\cdot)} = w_i(t); \quad \frac{\partial F_i[u(\cdot), \alpha]}{\partial \alpha} = a_i$$

осуществляется обычным образом и требует трехкратного интегрирования задачи Коши (с данными при $t=T$) для системы

$$-\dot{\psi} = A^*\psi + uB^*\psi. \quad (5)$$

Для вычисления, например, производной F_2 данные Коши для $\psi(T)$ имеют вид

$$\psi_1(T) = 0; \quad \psi_2(T) = 1; \quad \psi_3(T) = 0,$$

и после интегрирования (5) имеем

$$\delta F_2[\delta u(\cdot), \delta \alpha] = \int_0^T (\psi, Bx) \delta u(t) dt + (\psi_2(0) - 1) \delta \alpha, \quad (6)$$

т. е. $w_2(t) = (\psi(t), Bx(t)); a_2 = \psi_2(0) - 1$.

Решение задачи осуществлялось методом последовательной линеаризации (§§ 19—21). В качестве исходного управления задавалась функция $u(t) \equiv 1$, а значение α подбиралось так, чтобы

выполнялось условие $F_2=0$. То значение $x^3(T)$, которое в этом случае получалось, назначалось в качестве желаемого «уровня производства» D . После этого начинался собственно процесс оптимизации. После примерно 20 итераций была получена функция $u(t)$, изображенная на рис. 34; значение функционала F_0 стабилизировалось, условия $F_1=0$, $F_2=0$ были выполнены с большой точностью. Был проведен анализ полученного решения с целью выяснить, с чем связана стабилизация F_0 : с тем, что достигнута оптимальная (или почти оптимальная) ситуация, или с неэффективностью процесса минимизации. По существу, этот анализ состоял в проверке выполнения на полученной траектории условий принципа максимума, однако относительная простота задачи (в ее формулировке участвуют только три функционала, все они дифференцируемы по Фреше) позволяет придать анализу наглядную геометрическую форму: будут построены конусы достижимости K_F (см. § 5) для численно оптимальной и (для сравнения) для исходной траекторий. Начнем с анализа исходной, неоптимальной траектории. Прежде всего опишем конус возможных вариаций управления K_u . Так как $u(t) \equiv 1$, то из условия $0 \leq u(t) + \delta u(t) \leq 1$ получаем (для малых δu)

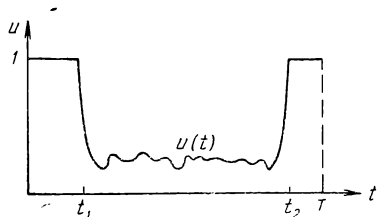


Рис. 34.

$$K_u: \quad \delta u(t) \leq 0; \quad 0 \leq t \leq T. \quad (7)$$

Таким образом, конус K_u состоит из всех неположительных на $[0, T]$ функций. Однако в задаче есть еще параметр α , его значения ограничены лишь физическим условием $\alpha \geq 0$, но поскольку $\alpha > 0$, то конус возможных значений $\delta \alpha$ есть числовая ось; обозначим ее K_α . Рассмотрим отображение конуса $K_u \times K_\alpha$ в трехмерное пространство смещений $\{\delta F_0, \delta F_1, \delta F_3\}$, определяемое формулами теории возмущений

$$\delta F_i = \int_0^T w_i \delta u dt + a_i \delta \alpha, \quad \delta u(\cdot) \in K_u, \quad \delta \alpha \in K_\alpha. \quad (8)$$

Образ $K_u \times K_\alpha$ есть конус достижимости K_F . Он, точнее, только та его часть, которая является образом K_u , без учета $\delta \alpha$, изображен на рис. 35. Поясним его. Рассмотрим зависящее от параметра t семейство лучей

$$\{\lambda \omega_0(t), \lambda w_1(t), \lambda w_2(t)\} \quad (9)$$

и его пересечение с плоскостью $\delta F_0=1$ (т. е. определим $\lambda(t)$ из условия $\lambda(t)w_0(t)=1$). На рис. 35 для некоторой покрывающей $[0, T]$ сетки t_n изображены точки

$$\{\lambda(t_n)w_1(t_n); \lambda(t_n)w_2(t_n)\}, \quad n=1, 2, \dots, N,$$

Совокупность этих точек описывает кривую I , которую следует представлять расположенной на плоскости $\delta F_0=1$. Содержательный смысл этой кривой прост. Пусть управление возмущается

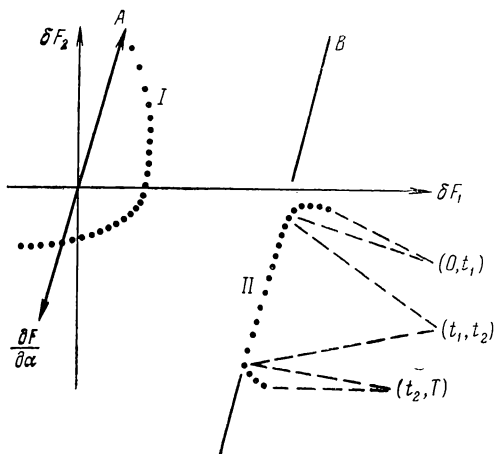


Рис. 35.

некоторой величиной δu , отличной от нуля только в малой окрестности точки t_n . Тогда следствием этого возмущения будет смещение точки δF на величину, пропорциональную вектору $\{w_0(t_n)\delta u; w_1(t_n)\delta u; w_2(t_n)\delta u\}$, т. е. δF сместится из точки $\{0, 0, 0\}$ по прямой, соединяющей начало координат с точкой кривой I , соответствующей значению параметра t_n . В данной ситуации $\lambda(t) > 0$, и если $\delta u(\cdot) \in K_u$, т. е. $\delta u(t) \leq 0$, то смещение может произойти только вниз по упомянутой прямой. Комбинируя подобные возмущения в окрестности всех точек I , получим выпуклую оболочку таких элементарных смещений δF . Легко представить себе строение этой оболочки: нужно поместить кривую I в плоскость $\delta F_0=-1$, взять ее выпуклую оболочку и ее точки соединить с началом координат. Это и будет конус достижимости K_F . Он содержит направление $\{-1; 0; 0\}$, следовательно, данная траектория неоптимальна. Учитывая еще $\delta \alpha$, мы только расширим K_F .

Такое же построение осуществим и для оптимального $u(t)$. Структура этой функции такова: отрезок $[0, T]$ разбит на три части точками t_1, t_2 : $0 < t_1 < t_2 < T$; причем

$$u(t) \begin{cases} = 1 & \text{при } t \in (0, t_1) \cup (t_2, T), \\ \in (0, 1) & \text{при } t \in (t_1, t_2). \end{cases} \quad (10)$$

Отсюда следует и структура конуса допустимых вариаций управления K_u :

$$\delta u(\cdot) \in K_u: \begin{cases} \delta u(t) \leq 0 & \text{при } t \in (0, t_1) \cup (t_2, T), \\ \delta u(t) \text{ произвольна} & \text{при } t \in (t_1, t_2). \end{cases} \quad (11)$$

Его образ в отображении (8) представим кривой Π в плоскости $\delta F_0 = 1$ (и в этом случае $\lambda(t) > 0$). Кривая Π состоит из трех частей, соответствующих интервалам времени $(0, t_1)$, (t_1, t_2) , (t_2, T) . Рассмотрим часть конуса K_F , соответствующую допустимым вариациям $\delta u(t)$, отличным от нуля только на (t_1, t_2) . Она описывается формулой

$$K_F^2: \quad \delta F = \int_{t_1}^{t_2} w(t) \delta u(t) dt, \quad \delta u(t) \text{ произвольна.} \quad (12)$$

Так как соответствующая часть кривой Π с точностью нанесения на график лежит на прямой B (в плоскости $\delta F_0 = 1$), то $K_F^{(2)}$ есть плоскость Π , проходящая через начало координат и прямую B . Части кривой Π , соответствующие интервалам $(0, t_1)$ и (t_2, T) , расположены правее B , но так как здесь возможны только вариации $\delta u \leq 0$, то конус

$$\delta F = \int_0^T w(t) \delta u(t) dt, \quad \delta u(\cdot) \in K_u, \quad (13)$$

становится полупространством, лежащим выше плоскости Π (в том смысле, что точка $\{1; 0; 0\}$ этому полупространству принадлежит, а точка $\{-1; 0; 0\}$ — нет). Однако мы должны еще расширить конус (13) до конуса (8), включив и вариацию δa . Структура системы (1) такова, что $a_0 = 0$, а вектор $\{a_1, a_2\}$ изображен на рис. 35. Всевозможные вариации δa порождают смещение δF вдоль прямой A , лежащей в плоскости $\delta F_0 = 0$. Эта прямая с хорошей точностью параллельна прямой B , следовательно, она лежит в плоскости Π и не приводит к расширению конуса (13). Это есть не что иное, как выполнение условия трансверсальности. Таким образом, конус достижимости K_F не содержит луча $\{-1, 0, 0\}$, т. е. траектория удовлетворяет принципу максимума. Конечно, в этом анализе полученная в расчетах картина идеализирована, мы отвлеклись от некоторых погрешностей (в частности, точки кривой Π ,

соответствующие (t_1, t_2) , разумеется, точно на прямую B не помещаются, параллельность A и B тоже не абсолютна). Однако эти погрешности настолько малы, что на чертеже не заметны. Конечно, сама функция $u(t)$ (см. рис. 34) на участке (t_1, t_2) найдена не очень точно. Изломы этой функции являются счетными эффектами и с существом дела не связаны. Однако, это не является очень уж большим недостатком численного решения. Практическое использование подобных результатов состоит обычно из следующих двух элементов: прежде всего выясняется, каков экономический эффект перехода от некоторого, найденного эмпирически, управления к оптимальному, и стоит ли последнее использовать. Если переход к оптимальному управлению признается целесообразным, возникает задача его аппроксимации теми или иными техническими средствами. Ведь точное отслеживание на реальном аппарате произвольной функции $u(t)$, как правило, или невозможно, или требует слишком сложной и дорогой добавочной аппаратуры. К счастью, нет необходимости в точном воспроизведении $u(t)$. В данном случае управление

$$u(t) = \begin{cases} 1 & \text{при } 0 \leq t < t_1, \\ \alpha & \text{при } t_1 < t < t_2, \\ 1 & \text{при } t_2 < t < T \end{cases} \quad (14)$$

дает почти тот же эффект, что и точное. Разумеется, параметры конструкции (14) должны быть соответственно подобраны решением, например, исходной вариационной задачи в классе функций (14). Однако, как сама конструкция (14), так и довольно точные приближения к оптимальным параметрам (t_1, t_2, α) , легко определяются по изображенной на рис. 34 численно оптимальной функции $u(t)$.

§ 33. Выбор оптимальных композиций защиты от излучения

Здесь будет описана задача оптимального управления, связанная с проблемой выбора наилучшего состава слоя защиты, отделяющего ядерный реактор от рабочих помещений. Ядерный реактор является мощным источником нейтронов и γ -лучей. Он отделен от помещений достаточно толстым слоем защиты, имеющей назначение ослабить поток первичного излучения до безопасного уровня. Однако ситуация осложняется тем, что сам этот защитный слой, поглощая нейтроны и γ -кванты, становится источником излучения (вторичного). Защитный слой обычно состоит из нескольких веществ, каждое из которых объединяет в себе как достоинства, так и недостатки. Так, вещество может хорошо поглощать быстрые нейтроны, являясь в то же время мощным

источником вторичных γ -квантов и т. д. Математическая постановка задачи будет приведена в несколько упрощенной форме, но это упрощение не связано с принципиальными вопросами.

Численное решение задачи оптимизации защиты проводилось для следующей модели. Защитный слой представляет собой отрезок $0 \leq t \leq T$ (по традиции мы используем обозначение t для независимой координаты, хотя физически это, конечно, не время, а пространственная координата). Защитный слой заполнен смесью четырех веществ, «управляющая функция» $u(t) = \{u_1(t), u_2(t), u_3(t), u_4(t)\}$ описывает их относительные концентрации в данной точке t . Разумеется, должны быть выполнены условия $u_i \geq 0$, $u_1 + u_2 + u_3 + u_4 = 1$. В расчетах концентрация u_4 исключалась: $u_4 = 1 - u_1 - u_2 - u_3$, и управление было трехмерным $u(t) = \{u_1, u_2, u_3\}$, удовлетворяющим ограничению

$$u(t) \in U: \quad u_1 \geq 0, \quad u_2 \geq 0, \quad u_3 \geq 0, \quad u_1 + u_2 + u_3 \leq 1. \quad (1)$$

Прохождение нейтронов через защитный слой описывалось системой уравнений

$$\begin{aligned} dx^1/dt &= A_{12}(u)x^2, \\ dx^2/dt &= A_{21}(u)x^1. \end{aligned} \quad (2)$$

Здесь x^1, x^2 — 5-мерные векторы; физически координаты $x^1(t)$ описывают поток нейтронов в данной точке t , причем энергетический спектр представлен пятью группами. Система (2) замыкается краевыми условиями

$$\Gamma(x) = 0: \quad x^1(0) - S = 0; \quad x^2(T) = 0, \quad (3)$$

где $S = \{s_1, s_2, \dots, s_5\}$ — поток нейтронов, падающий на внутреннюю границу защиты. Что касается матриц $5 \rightarrow 5$ A_{12}, A_{21} — это матрицы специфической структуры (нижние треугольные), их элементы заданным образом зависят от концентраций $u(t)$. Большей частью зависимость A от u была линейной:

$$A[u(t)] = A^0(t) + A^1 u_1(t) + A^2 u_2(t) + A^3 u_3(t) + A^4 u_4(t). \quad (4)$$

Составляющая A^0 добавлена в связи с тем, что защитный слой мог содержать и обязательные составляющие, входящие, например, в элементы несущих конструкций. Заметим, что численное решение задачи (2), (3) достаточно сложно, так как среди решений (2) есть как сильно растущие, типа $e^{\lambda t}$, так и сильно падающие экспоненты типа $e^{-\lambda t}$, причем $\lambda T \approx 30-40$ (а если в состав защиты входит бор в заметных концентрациях, то мы сталкиваемся и с ситуациями $\lambda T \approx 100$). Однако к тому времени (1965 г.), когда была начата работа по численному решению задачи оптимизации, расчет защиты (в принятой здесь модели) был уже освоен и принципиальных трудностей не содержал. Решение краевой задачи (2), (3) легко

осуществляется последовательными прогонами, а для обеспечения точности разностной аппроксимации было достаточно использовать шаг dt , удовлетворяющий лишь условию $|\lambda dt| \ll 1$. (В наших расчетах $dt \sim T/500$.) Кроме нейтронов, нужно было учитывать и γ -кванты, они также описывались пятью энергетическими группами, однако нас интересовала «биологическая опасность» γ -излучения, т. е. скалярная величина

$$\gamma^* = \sum_{i=1}^5 c_i \gamma_i(T), \quad (5)$$

где c_i — «вес» i -й группы γ -квантов. Величина γ^* есть функционал от состава (управления) $u(t)$, вычисляемый по формуле

$$\begin{aligned} \gamma^*[u(\cdot)] \equiv & \mathcal{F}_0 \left[\int_0^T a(u) dt, \int_0^T b(u) dt, \Gamma_0 \right] + \\ & + \int_0^T \mathcal{F}_1 \left[x^1(t), u(t), \int_t^T a(u) d\tau, \int_t^T b(t) d\tau \right] dt. \end{aligned} \quad (6)$$

Здесь $a(u)$, $b(u)$ — заданные 5-мерные функции от u , а $\int_t^T a[u(\tau)] d\tau$,

$\int_t^T b[u(\tau)] d\tau$ — суть величины, связанные с так называемыми «оптическими толщинами» слоя защиты $[t, T]$ относительно γ -квантов разных энергий. \mathcal{F}_0 и \mathcal{F}_1 — заданные функции своих аргументов, точный вид их здесь не так уж важен. Наконец, Γ_0 характеризует поток первичного γ -излучения, падающего на внутреннюю границу защиты. Для придания задаче стандартной формы введем 5-мерные фазовые компоненты $x^3(t)$ и $x^4(t)$, удовлетворяющие уравнениям

$$\begin{aligned} \frac{dx^3(t)}{dt} &= -a[u(t)]; \quad x^3(T) = 0, \\ \frac{dx^4(t)}{dt} &= -b[u(t)]; \quad x^4(T) = 0. \end{aligned} \quad (7)$$

Тогда и функционал (6) запишется в стандартной форме

$$\gamma^*[u(\cdot)] \equiv \mathcal{F}_0[x^3(0), x^4(0), \Gamma_0] + \int_0^T \mathcal{F}_1[x^1(t), u(t), x^3(t), x^4(t)] dt. \quad (8)$$

Кроме того, в постановках вариационных задач использовались

функционалы

$$F_0[u(\cdot)] \equiv \int_0^T \rho[u(t)] dt \quad (\text{вес защиты}),$$

$$F_1[u(\cdot)] \equiv (d, x^1(T)) \quad (\text{доза нейтронов}).$$

Здесь d — заданный 5-мерный вектор, компоненты которого учитывают биологическую опасность нейтронов соответствующей энергии.

Стандартная вариационная задача состояла в следующем. Найти $u(t)$ из условий:

$$\begin{aligned} 1) & \min_{u(\cdot)} F_0[u(\cdot)]; \\ 2) & F_1[u(\cdot)] \leq D_n; \\ 3) & \gamma^*[u(\cdot)] \leq D_\gamma; \\ 4) & u(t) \in U. \end{aligned} \quad (9)$$

Численное решение этой задачи осуществлялось методом последовательной линеаризации (§§ 19—21). Проводились массовые расчеты для разных потоков S и Γ_0 , и для разных наборов веществ, из которых составлялась защита. Некоторые результаты описаны в [1], [59], [73].

Управление $u(t)$ искалось в классе кусочно постоянных функций на сетке, содержащей 64 интервала. В некоторых расчетах искомыми элементами управления являлись не только концентрации веществ u , но и физические толщины интервалов постоянства u . Другими словами, от уравнений (2) мы переходили к уравнениям в формальном времени τ ,

$$\frac{dx^1}{d\tau} = v(\tau) A_{12}(u) x^2; \quad \dots \quad \frac{dt}{d\tau} = v(\tau); \quad 0 \leq \tau \leq 1, \quad (10)$$

и $v(\tau)$ становилось дополнительной компонентой управления. Размерность фазового пространства $\{x^1, x^2, x^3, x^4\}$ в задаче равна 20. Расчеты проводились на машине с быстродействием $5 \cdot 10^4$ операций в секунду и с оперативной памятью 2^{12} ячеек. Стандартный расчет (25—30 итераций) занимал около 30 минут машинного времени, причем последние 5—10 итераций проводились при фактически стабильном значении минимизируемого функционала F_0 , первые 5—10 итераций составляло решение «терминальной» задачи — находилось управление, удовлетворяющее ограничениям по допустимым дозам (условиям (9.1) и (9.2)). В процессе решения задачи условие $u(t) \in U$ никогда не нарушалось. Оптимальность найденных композиций защиты контролировалась проверкой выполнения принципа максимума. В данной задаче все функционалы дифференцируемы по Фреше, и проверка сводилась к анализу конуса смещений K_F (см. § 5).

Опишем подробнее один из характерных примеров [59]. Рассматривалась цилиндрическая одномерная защита, состоящая из воды, железа и бора, на которую падает поток нейтронов спектра деления, равный $10^{10} \frac{1}{\text{см}^2 \cdot \text{сек}}$, и поток γ -квантов, возникающий при делении урана, равный $10^{10} \frac{1}{\text{см}^2 \cdot \text{сек}}$.

Была поставлена задача найти такое распределение компонент защиты, которое минимизирует ее вес при выполнении заданных ограничений доз нейтронов и γ -квантов на поверхности защиты. Расчет пространственно-энергетического распределения потока нейтронов проводился пятигрупповым методом в P_1 -приближении кинетического уравнения (с заданием ведущей группы). Интервалы сетки для управления (t_{n-1}, t_n) будем нумеровать индексом n . На каждом таком интервале определен элементарный «конус» возможных (совместимых с условием $u_n + \delta u_n \in U$) вариаций управления K_n . Этот конус строится следующим образом. Выбираются векторы $\{e_{n,1}; e_{n,2}; e_{n,3}\}$ и соответствующие пары величин $\{s_{n,i}^-, s_{n,i}^+\}$, $i=1, 2, 3$, так, что при любых $s_{n,i}^- \leq s_{n,i} \leq s_{n,i}^+$

$$u_n + \sum_{i=1}^3 s_{n,i} e_{n,i} \in U, \quad n=1, \dots, N. \quad (11)$$

Вычисление «базиса» $e_{n,i}$ и границ s^-, s^+ должно учитывать геометрию области U и положение точки u_n в U (см. в связи с этим рис. 16 § 19). Следствием вариации управления (11) является смещение точки трехмерного пространства $F = \{F_0, F_1, \gamma\}$ в положение

$$F + \delta F = F + \sum_{n=1}^N \sum_{i=1}^3 s_{n,i} h_{i,i}, \quad (12)$$

где $h_{n,i} = \{h^0, h^1, h^2\}_{n,i}$ — вектор, вычисление компонент которого требует, в соответствии с общей схемой §§ 19—21, трехкратного решения сопряженного уравнения (точнее, двухкратного, так как в силу простоты выражения для веса F_0 компонента h_0 вычисляется непосредственно). Итак, конус (точнее, многогранник) вариаций $K_n = \prod_{i=1}^3 K_{n,i}$ (конструкция K_n описана формулой (11)) отображается, в соответствии с (12), в множество достижимости K_F , и нужно проанализировать K_F : если K_F пересекается с конусом запрещенных смещений K_3 в трехмерном пространстве, композиция не может быть оптимальной. Так как условия задачи имеют характер неравенств $F_1 \leq D_n$, $\gamma^* \leq D_\gamma$, то K_3 есть выпуклая оболочка векторов $(-1, 0, 0)$, $(0, -1, 0)$, $(0, 0, -1)$; в оптимальной ситуации ни один из этих векторов не должен лежать внутри

K_F . Можно считать, что для границ s^- , s^+ возможны лишь два варианта:

- 1) $s_n^- < 0, \quad s_n^+ > 0;$
- 2) $s_n^- = 0; \quad s_n^+ > 0;$

третий случай $s_n^- < 0, s^+ = 0$ сведем ко второму, изменив знак у соответствующего h на обратный. Векторы h , для которых $s^- < 0$ и $s^+ > 0$, будем называть *свободными*, остальные — *односторонними*. Последние в свою очередь подразделяются на две группы по знаку компоненты h^0 ; здесь мы будем называть их соответственно *положительными* и *отрицательными*. В этих терминах будет удобно описать конструкцию конуса K_F . Этот конус является выпуклой оболочкой совокупности порожденных векторами h лучей, причем каждый свободный вектор порождает в трехмерном пространстве прямую (два луча) $\{sh, s \text{ — любое}\}$, а каждый односторонний h — лишь полупрямую (луч) $\{sh, s \geq 0\}$. Удобно изображать эти прямые точками их пересечений с плоскостями $\delta F_0 = 1$ и $\delta F_0 = -1$. Прямые, порожденные свободными векторами, пересекают обе эти плоскости (но на рисунке мы будем изображать точку пересечения лишь с одной плоскостью, в зависимости от знака h^0); лучи, порожденные положительными (отрицательными) односторонними h , пересекают только плоскость $\delta F_0 = 1$ ($\delta F_0 = -1$). Именно эти точки и будут изображаться на рисунках в плоскости $\{\delta F_1, \delta \gamma^*\}$.

В качестве исходной композиции была выбрана защита, состоящая из чередующихся слоев воды и железа. Построенный в этой ситуации конус K_F изображен на рис. 36, причем, так как точки u_n занимают угловые позиции в области U , здесь имеются только односторонние векторы. Пересечения соответствующих лучей с плоскостью $\delta F_0 = 1$ обозначены «+», пересечения с плоскостью $\delta F_0 = -1$ обозначены «•». Нужно еще учесть, что для разных групп векторов h пришлось использовать разные масштабы: если бы масштаб был единым, разброс точек был бы еще большим. Анализ этой ситуации с очевидностью приводит к выводу, что конус K_F заполняет все трехмерное пространство. На рис. 37 изображены концентрации компонент защиты, найденные в результате решения вариационной задачи. С точки зрения уменьшения веса железа выгоднее передвинуть во внутренние области защиты. Однако, так как при этом происходит увеличение генерации захватного γ -излучения, в эти области приходится добавлять большие по сравнению с наружными областями количества бора. Вес одного погонного сантиметра защиты в исходной композиции равен 290 кг, для оптимального варианта 190 кг. Структура конуса достижимости K_F в оптимальной ситуации выясняется с помощью рис. 38. На нем изображены следы лучей, порожденных свободными векторами h (изображены «•»). Эти следы расположены вблизи двух

параллельных прямых, одна из которых (A) лежит в плоскости $\delta F_0=1$, другая (B) — в плоскости $\delta F_0=-1$. Отвлекаясь от погрешностей вычислений и считая все эти точки расположенными

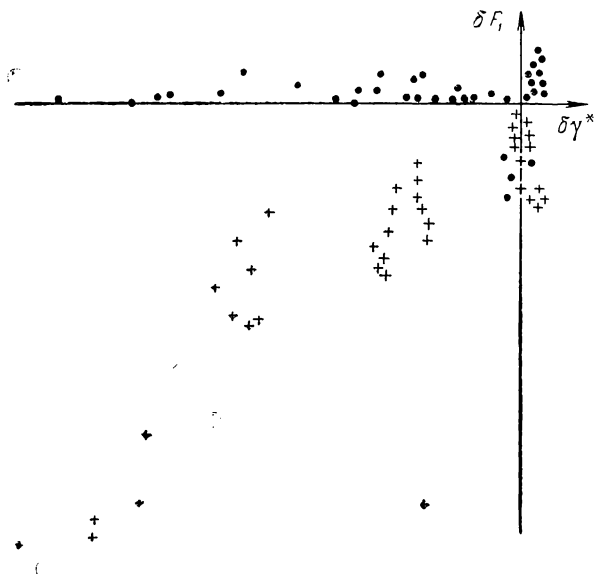


Рис. 36.

на упомянутых прямых, в качестве их выпуклой оболочки получим плоскость, проходящую через начало координат и через обе

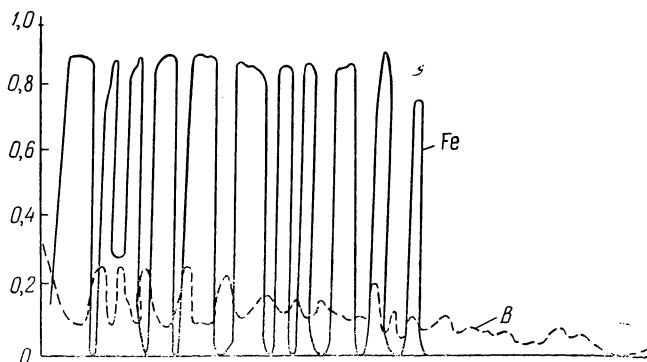


Рис. 37.

прямые. Кроме того, имеются еще односторонние векторы h , порождающие пересечения с плоскостью $\delta F_0=1$ («+») и с плоскостью

$\delta F_0 = -1$ («0»). Учитывая и эти лучи, получим в качестве K_F полупространство, лежащее выше построенной уже плоскости. Выше в том смысле, что, как нетрудно видеть, векторы $(-1; 0; 0)$, $(0; -1; 0)$ и $(0; 0; -1)$ лежат вне этого полупространства. Это и есть критерий оптимальности композиции (принцип максимума). Разумеется, наши рассуждения нестроги, мы несколько идеализировали действительное расположение точек на рис. 38. Однако это искажение действительной картины невелико, особенно если сравнить его с разбросом точек на рис. 36. Если бы рис. 36 и 38 были даны в одном и том же масштабе, разброс «+» около прямых A и B стал бы совсем незначительным.

Выше уже было отмечено, что, строго говоря, K_F заполняет все пространство, и продолжение итераций, казалось бы, должно приводить к улучшению управления. В действительности этого не происходит. Точнее, если управление варьируется на величину $\approx 5-10\%$ от u (примерно такая величина вариации используется в процессе поиска), точки в окрестностях прямых на рис. 38 будут от итерации к итерации переходить с одной стороны прямой на другую. Если в конце расчета провести серию итераций с существенно меньшим шагом δu , можно добиться более точного попадания соответствующих свободным векторам точек на обе прямые. Здесь этого не делалось.

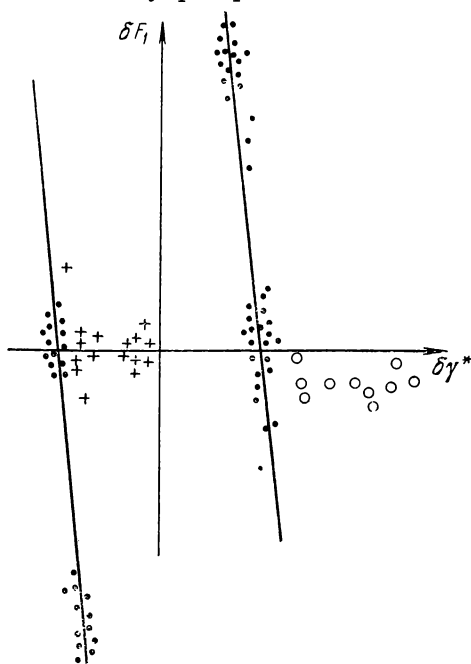


Рис. 38.

§ 34. Задача о стабилизации спутника

Уравнения движения управляемой системы следующие:

$$\begin{aligned} \dot{x}^1 &= A_1 x^2 x^3 + a_1 u^1, \\ \dot{x}^2 &= A_2 x^3 x^1 + a_2 u^2, \\ \dot{x}^3 &= A_3 x^1 x^2 + a_3 u^3, \\ 0 &\leq t \leq T, \quad x(0) = X_0 \end{aligned} \quad (1)$$

(X_0 , T заданы).

Минимизировать

$$F_0[u(\cdot)] \equiv \int_0^T \{|u^1(t)| + |u^2(t)| + |u^3(t)|\} dt \quad (2)$$

при условиях

$$x^1(T) = x^2(T) = x^3(T) = 0 \quad (F_i[u(\cdot)] \equiv x^i(T) = 0). \quad (3)$$

Мы не будем подробно объяснять физический смысл задачи. Ограничимся лишь указанием, что уравнения (1) описывают вращение твердого тела (спутника), снабженного тремя реактивными двигателями, F_0 есть расход топлива, условия (3) — суть условия отсутствия вращения (стабилизация). Эта задача была решена аналитически в [33], точное решение ее известно, и она представляет собой удобный методический пример. В дальнейшем была предпринята попытка численного решения задачи методом локальных вариаций [41]. Мы говорим лишь о попытке потому, что, как это станет ясным из дальнейшего, полученные в [41] численные результаты оказались очень грубыми. Наконец, в нашей работе [96] была показана возможность эффективного и весьма точного решения задачи методом проекции градиента.

Решение задачи методом локальных вариаций описано в [41] (это же решение воспроизведено в монографии [86]). Численное решение задачи описывалось сеточными функциями x_n^i ($n = 0, 1, \dots, N$), $u_{n+1/2}^i$, связанными конечно-разностными уравнениями (второго порядка точности):

$$\frac{x_{n+1}^i - x_n^i}{\tau} = A_1 \frac{x_n^2 + x_{n+1}^2}{2} \frac{x_n^3 + x_{n+1}^3}{2} + a_1 u_{n+1/2}^1, \quad n = 0, 1, \dots, N-1. \quad (4)$$

Такая же аппроксимация используется и для остальных уравнений.

Процесс минимизации интеграла

$$F_0 = \tau \sum_{n=0}^{N-1} \{|u_{n+1/2}^1| + |u_{n+1/2}^2| + |u_{n+1/2}^3|\} \quad (5)$$

по схеме метода локальных вариаций состоял в том, что поочередно варьировались компоненты $x_n^i \rightarrow x_n^i \pm h^i$, что в каждом элементарном акте варьирования приводит к изменению только значений $u_{n-1/2}^i$ и $u_{n+1/2}^i$. Та из вариаций, которая приводит к понижению (5), осуществлялась, т. е. происходило изменение сеточной траектории (элементарный акт приводит к изменению только одной компоненты x^i в одной только точке сетки t_n).

Первая задача. Система уравнений (см. [41])

$$\begin{aligned} \dot{x}^1 &= 0,166667u^1, & x^1(0) &= 24, \\ \dot{x}^2 &= -0,2x^3x^1 + u^2; & x^2(0) &= 16, \\ \dot{x}^3 &= 0,2x^1x^2 + 0,2u^3; & x^3(0) &= 16, \\ T &= 1. \end{aligned} \quad (6)$$

Точное значение $\min F_0 = 166,56$. В [41] отмечаются следующие особенности процесса решения этой задачи.

1. В качестве исходного управления бралась траектория $\{x^i(t) = x^i(0)(1 - t/T)\}$.

Эта траектория в [41] характеризуется как локальный минимум задачи: локальные вариации оставляют ее неизменной, так как переходы $x_n^* \rightarrow x_n^* \pm h$ не приводят к уменьшению (5) (при любом h).

2. Функционал (2) был «регуляризован», т. е. заменен на

$$F_0[u(\cdot), \varepsilon] \equiv \int_0^T \sum_{i=1}^3 \{|u^i(t)| + \varepsilon |u^i(t)|^2\} dt, \quad (7)$$

и добавлен процесс постепенного уменьшения ε до нуля. Следующая таблица (заимствована из [41]) показывает, какие значения F_0 были получены при разных ε (причем решение, полученное для ε_k , служило начальным приближением при ε_{k+1}).

ε	0,1	0,05	2^{-10}	2^{-18}	0
F_0	171,88	170,36	169,55	169,43	169,42

Значение F_0 на исходной траектории равно 370,16. В [41] не приведено данных, позволяющих судить об объеме вычислительной работы, связанной с получением этого приближенного решения.

3. Как и в точном решении, в приближенном $u^3(t) \equiv 0$. Что касается $u^1(t)$ и $u^2(t)$, то они с точными не сравниваются, поскольку, как было выяснено еще в [33], решение вариационной задачи неединственно.

Вторая задача.

$$\begin{aligned} \dot{x}^1 &= \frac{1}{3} x^2 x^3 + 100u^1, & x^1(0) &= 200, \\ \dot{x}^2 &= -x^3 x^1 + 25u^2, & x^2(0) &= 30, \\ \dot{x}^3 &= x^1 x^2 + 100u^3, & x^3(0) &= 40, \quad T = 1. \end{aligned} \quad (8)$$

В качестве начальной траектории и здесь бралась прямая (в пространстве $\{x^1, x^2, x^3, t\}$), соединяющая точку $\{200; 30; 40; 0\}$ с точкой $\{0; 0; 0; T\}$. Расчеты дали траекторию со значением $F_0 = 3,55$. В [41] приведено и точное значение $\min F_0 = 3,5$, взятое из результатов аналитического решения задачи в [33]. (Это значение, как мы увидим, ошибочное. В действительности, $\min F_0 = 2,5075$).

Заметим, что задача очень благоприятна для решения ее методом локальных вариаций: в ней отсутствуют ограничения $u(t) \in U$,

и размерности x и u совпадают, так что любая траектория $x(t)$ может быть реализована управлением, вычисляемым по формулам

$$u^1 = \frac{1}{a_1} [\dot{x}^1 - A_1 x^2 x^3].$$

В большинстве прикладных задач размерность управления меньше размерности x , и с этим связаны определенные вычислительные трудности (см. §§ 15, 16).

Решение задачи методом проекции градиента было проведено в целях сравнения двух методов и иллюстрации возможностей метода проекции градиента. Результаты опубликованы в [96]; здесь они воспроизводятся. Заметим, что задачи (6) и (8) решались и методом последовательной линеаризации (§ 19), но результаты мы приводить не будем, так как они практически те же самые.

Прежде всего мы перейдем к эквивалентной постановке задачи с тем, чтобы минимизируемый функционал не содержал знака модуля и был бы дифференцируем по Фреше. Правда, при этом появляются условия типа $u(t) \in U$. Введем вместо трех управляющих функций $u^i(t)$ шесть других $v^i(t)$, $w^i(t)$, $i=1, 2, 3$, связанных с $u^i(t)$ соотношением

$$u^i(t) = v^i(t) + w^i(t), \quad i = 1, 2, 3, \quad (9)$$

и ограниченных условиями

$$v^i(t) \geq 0, \quad w^i(t) \leq 0, \quad i = 1, 2, 3. \quad (10)$$

Представление $u^i(t)$ в виде (9) осуществляется так:

$$v^i = \begin{cases} u^i, & \text{если } u^i \geq 0; \\ 0, & \text{если } u^i < 0; \end{cases} \quad w_i = \begin{cases} 0, & \text{если } u^i > 0, \\ u^i, & \text{если } u^i \leq 0. \end{cases} \quad (11)$$

Таким образом,

$$|u^i(t)| = v^i(t) - w^i(t). \quad (12)$$

Теперь система (1) приобретает вид

$$\begin{aligned} \dot{x}^1 &= A_1 x^2 x^3 + a_1 v^1 + a_1 w^1, \\ \dot{x}^2 &= A_2 x^3 x^1 + a_2 v^2 + a_2 w^2, \\ \dot{x}^3 &= A_3 x^1 x^2 + a_3 v^3 + a_3 w^3, \\ 0 &\leq t \leq T; \quad x(0) = X_0, \end{aligned} \quad (1^*)$$

и задача состоит в определении $\{v(\cdot), w(\cdot)\}$ из условий

$$\min \int_0^T \sum_{i=1}^3 [v^i(t) - w^i(t)] dt \quad (\min F_0[v(\cdot), w(\cdot)]). \quad (2^*)$$

Разумеется, учитываются и условия (3), (10). Расчеты проводились по схеме, изложенной в §§ 18—21. Вводилась сетка $0 = t_0 < t_1 <$

Т а б л и ц а 1

I		II		III	
ν	F_0	ν	F_0	ν	F_0
0	371,34	0	371,35	0	371,34
1	333,26	1	333,05	1	303,80
2	303,27	2	302,56	2	265,19
3	278,75	3	278,97	3	238,14
4	257,40	4	258,46	4	212,62
5	239,47	5	240,45	5	193,28
6	224,37	6	224,91	6	178,75
7	211,80	7	212,24	7	169,85
8	200,01	8	200,98	8	166,71
9	190,73	9	191,37	9	166,61
10	183,20	10	183,72		
11	177,21	11	177,95		
12	172,95	12	173,38		
13	169,79	13	170,20		
14	167,78	14	167,99		
15	166,69	15	166,82		
16	166,61	16	166,63		

$< t_2 < \dots t_N = T$, $t_n = n\tau$, $\tau = T/N$, управление искалось в классе кусочно постоянных функций

$$v^i(t) = v_{n+1/2}^i, \quad w^i(t) = w_{n+1/2}^i, \quad \text{при } t_n < t < t_{n+1}.$$

Таким образом, горизонтальная размерность задачи квадратического программирования (§ 49) (или линейного программирования (§ 48)) равна $6N$ (расчеты проводились с $N=50$ и с $N=100$), вертикальная размерность $m=3$. Табл. 1 иллюстрирует процесс решения первой задачи, ν есть номер итерации, F_0 — значение функционала на данной итерации. В качестве исходной траектории, как и в [41], бралось управление, соответствующее линейным $x^i(t)$. В первом расчете $N=50$, вариации компонент управления $|\delta v^i|$, $|\delta w^i|$ были ограничены числами 20, 10, 30 (для $i=1, 2, 3$ соответственно). В процессе решения задачи условия $x^i(T)$ были выполнены с абсолютной погрешностью, не превышающей 0,02. Второй расчет отличался от первого только значением $N=100$. Время решения задачи возросло в два раза. Наконец, в третьем расчете, при $N=50$, были разрешены большие значения вариаций $|\delta v^i|$, $|\delta w^i|$; они были ограничены значениями 40, 20, 60. Время решения задачи сократилось почти вдвое, точность выполнения условий $x^i(T)=0$ осталось той же, что и в первом расчете. Видимо, возможно и дальнейшее увеличение допустимых значений $|\delta v|$, $|\delta w|$, что приводит к дальнейшему сокращению времени решения

задачи. Сравним теперь наши расчеты с расчетами в [41]. Что касается затрат машинного времени, то в наших расчетах одна итерация состоит из следующих вычислений:

- 1) интегрирование «прямой» системы (1*);
- 2) трехкратное интегрирование сопряженной системы;
- 3) решение задачи квадратичного (линейного) программирования.

В целом одна итерация по затратам времени соответствует, примерно, пятикратному интегрированию прямой системы. Заметим, что эта система (как и сопряженные) интегрировалась не с шагом сетки $\tau = T/N$, а с меньшим, обеспечивающим высокую точность вычисления значений $x^i(t)$. В [41] нет данных, которые позволили бы составить хоть какое-то представление о трудоемкости решения задачи методом локальных вариаций. Однако сравнение точности полученных решений можно произвести. В [41] найдено решение с $F_0 = 169,42$, ошибка 2,87 составляет $\sim 1,7\%$ от $F_0 = 166$. В наших расчетах ошибка не превосходит 0,07, т. е. 0,04%. В действительности относительная погрешность расчетов больше. Дело в том, что величина F_0 состоит из двух частей:

$$F_0 = \int_0^T |u^1(t)| dt + \int_0^T |u^2(t)| dt = 144 + 22,56 = 166,56 \quad (13)$$

$$(u_3(t) \equiv 0).$$

Но уравнение для x^1 в силу $A_1 = 0$ очень просто, и из условий $x^1(T) = 0$ и $u^1(t) \leq 0$ следует, что первое слагаемое (144) будет найдено точно, какой бы ни была функция $u^1(t)$ (а в данной задаче имеется семейство решений, дающих одно и то же минимальное значение F_0 , так что $u^1(t)$ определяется с большой степенью неопределенности). Таким образом, вся ошибка численного решения связана с ошибкой во втором слагаемом, и относительная погрешность в нем составляет $\sim 12,5\%$ для метода локальных вариаций и $\sim 0,3\%$ в наших расчетах. В [41], [86] исходная траектория характеризуется как точка локального минимума вариационной задачи. Это, как показали наши расчеты, неверно. Легко проверить (предоставим это читателю), что исходная траектория является стационарной точкой метода локальных вариаций: принятая в этом методе техника варьирования траектории действительно не приводит к изменению значения функционала. Но это есть следствие дефекта метода, а не особенность данной траектории. Ведь если бы мы имели дело с локальным минимумом задачи, то и наш метод не позволил бы эту траекторию проварьировать: как и всякий реализуемый метод, он является методом поиска лишь локального минимума. Поэтому замену функционала (2) на функционал

(7), позволившую в расчетах методом локальных вариаций «сдвинуться» с места, едва ли правильно называть регуляризацией. В самом деле, о регуляризации можно говорить в том случае, когда имеется мощное семейство траекторий с одинаковым (или почти одинаковым) значением F_0 (равным $\min F_0$), и тогда малая добавка к F_0 (переход к (7)) позволяет (в принципе) из этого множества траекторий выбрать некоторую, предпочтительную по дополнительному признаку (в данном случае предпочтение отдавалось бы траектории с меньшим значением $\int \|u\|^2 dt$). В расчетах же [41], совсем другое дело: просто в задаче на минимум F_0 вида (7) исходная траектория не является стационарной для метода локальных вариаций. Заметим, что при $\varepsilon=0, 1$ (а именно это значение ε позволило от исходной траектории с $\int \sum |u^i| dt = 371$ перейти к траектории с $\int \sum |u^i| dt = 172$) значение «малой» добавки $\varepsilon \int \sum |u^i|^2 dt$ примерно в 15 раз больше основной части (7) $\int \sum |u^i| dt$. Успех этого приема связан с тем счастливым обстоятельством, что в данной задаче оптимальное по функционалу $\int \sum |u^i| dt$ управление мало отличается от оптимального по функционалу $\int \sum (|u^i| + |u^i|^2) dt$: и в том, и в другом случае $u^1(t) \approx -x^1(0)/a_1$ (см. рис. 6, 7 в [41]). В целом, как это видно из табл. 1, этот вариант задачи был довольно легким для численного решения. Вторая задача оказалась сложнее. Она была решена в несколько измененной по сравнению с [41] форме: во-первых, $T=0, 1$, а не 1, как в [41], и в качестве исходной траектории бралось решение задачи Коши (1) с $u^i(t) \approx 0$ (условия $x^i(T)=0$, разумеется, выполнены не были). Причины, побудившие к этим изменениям, будут ниже разъяснены. Табл. 2 дает представление о том, как происходит поиск: v — номер итерации, значение функционала F_0 , значения $x^i(T)$, предсказанные на предшествующей итерации на основе формул линейной теории возмущений, использующих функциональные производные $\partial x^i(T)/\partial u(\cdot)$, и значения $x^i(T)$, фактически полученные после интегрирования системы (1*). Процесс решения задачи заслуживает комментария. Расчет проводился при $N=100$, вариации $|\delta v^i|$, $|\delta w^i|$ были ограничены числами 2; 0,5; 0,5 для $i=1, 2, 3$.

1. Хорошо видны три этапа решения. Начальный этап (первые 17 итераций) — решение терминальной задачи. На этом этапе ищется управление, удовлетворяющее дополнительным условиям $x^i(T)=0$. Вариация $\{\delta v(\cdot), \delta w(\cdot)\}$ ищется с целью только уменьшения $\|x(T)\|$; значение δF_0 нас не интересует, а F_0 растет.

Т а б л и ц а 2

ν	F_0	$x^1(T)$ предск.	$x^1(T)$ факт.	$x^2(T)$ предск.	$x^2(T)$ факт.	$x^3(T)$ предск.	$x^3(T)$ факт.
0	0,2500		190,8	0	0,95		49,69
1	0,453	170,76	169,4	48,5	39,0	146,0	25,3
2	0,696	147,5	149,2	63,3	39,1	43,9	-19,8
3	0,954	130,9	131,0	20,8	4,54	-86,2	-41,3
5	1,448	87,52	89,1	-54,9	-34,9	-55,6	14,1
7	1,928	52,99	50,2	31,3	28,4	116,4	24,1
9	2,208	33,6	33,7	25,6	25,4	28,8	13,8
11	2,465	22,87	22,86	22,5	21,9	12,6	3,2
13	2,815	5,09	5,07	14,7	15,0	5,1	1,51
15	3,015	3,59	3,52	-5,89	6,12	2,67	1,93
17	3,267	0,228	0,228	0,23	0,13	0,21	-1,3
19	3,084	0,023	0,004	$5 \cdot 10^{-5}$	0,11	$2 \cdot 10^{-3}$	0,046
21	2,973	-0,015	-0,16	-0,005	0,007	-0,14	-0,14
23	2,872	-0,003	-0,003	-0,0088	-0,006	-0,05	-0,06
25	2,780	0	$5 \cdot 10^{-4}$	0	-0,1	0	0,08
27	2,700	0,015	-0,025	-0,030	-0,09	0,075	0,06
29	2,629	0,004	0,019	-0,003	-0,035	0,008	-0,008
35	2,531	0,023	0,022	-0,097	-0,106	-0,001	0,027
40	2,525	0	0,0042	0	0,0075	0	-0,037
45	2,521	$2 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$2 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$-4 \cdot 10^{-2}$
50	2,518	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	-0,019
55	2,516	0	$1,6 \cdot 10^{-3}$	0	$-3 \cdot 10^{-3}$	0	$-3 \cdot 10^{-3}$
60	2,515	$6 \cdot 10^{-5}$	$7 \cdot 10^{-3}$	-10^{-6}	$-1 \cdot 10^{-2}$	$2 \cdot 10^{-4}$	$-1 \cdot 10^{-3}$

Здесь могут возникнуть некоторые вопросы. В самом деле, на первой же итерации управление варьируется так, чтобы точка $x(T) = \{190,8; 0,95; 49,69\}$ перешла в точку $\{170,8; 48,5; 146\}$ (фактически $x(T)$ перешла в точку $\{169,4; 39,0; 25,3\}$). Видно, что основной целью является получить $x^1(T) = 0$, и ради этого допускается, например, значительное увеличение $x^3(T)$. Это самым тесным образом связано с используемой в наших расчетах нормировкой задачи. Дело в том, что функциональные производные $\partial x^2(T)/\partial u(\cdot)$, $\partial x^3(T)/\partial u(\cdot)$ примерно в 10 раз больше производной $\partial x^1(T)/\partial u(\cdot)$, и в «естественных» единицах измерения $x^i(T)$ величина $x^3(T) = 146$ становится числом ≈ 15 , малым сравнительно с $x^1(T) = 190$. Второй вопрос связан с плохой точностью линейного приближения для $x^2(T)$ и $x^3(T)$, в то время как для $x^1(T)$ точность линейного приближения высока: предсказанные и фактические значения $x^1(T)$ совпадают очень хорошо. На первый взгляд кажется, что плохое предсказание $x^2(T)$, $x^3(T)$ должно вынудить уменьшить шаги $|\delta v|$, $|\delta w|$, чтобы добиться лучшего. Однако этого делать не следует, так как в естественных единицах измерения величин $x^i(T)$ речь идет о несовпадении малых, сравнительно с $x^1(T)$, величин. В этом расчете ограничения на величины вариаций управления

оставались неизменными. Нетрудно понять причину различия между $x^1(T)$ и $x^2, x^3(T)$. Для этого используем полезные и в дальнейшем первые интегралы системы (1) при $u(t) \equiv 0$:

$$\begin{aligned} [x^2(t)]^2 + [x^3(t)]^2 &= 50^2; \\ [x^1(t)]^2 + \frac{1}{3} [x^2(t)]^2 &= 200^2 + \frac{1}{3} \cdot 30^2. \end{aligned} \quad (14)$$

Из этой системы видно, что $|x^2(t)| \leq 50$, и поэтому $x^1(t)$ испытывает лишь небольшие колебания около значения 200. Что касается $x^2(t)$ и $x^3(t)$, то они изменяются аналогично функциям

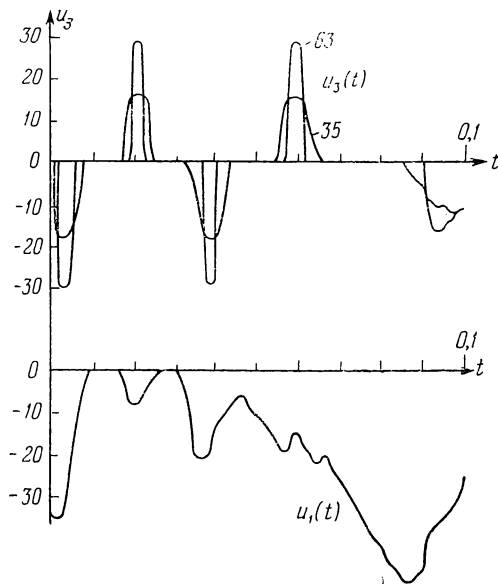


Рис. 39.

$\sin 200t, \cos 200t$, для них $x^1(t)$ играет роль мгновенной частоты, даже не очень большое изменение которой за большое для частоты 200 время 0,1 приводит к большому изменению $x^2(T), x^3(T)$. После того как при $\nu \approx 17$ функция $x^1(t)$ более или менее стабилизировалась, поведение $x^2(T), x^3(T)$ при тех же вариациях управления стало много лучше предсказываться линейной теорией возмущений.

Второй этап решения — основная оптимизация (это итерации от 17-й до 35-й). На этом этапе, при $x^i(T) \approx 0$, F_0 достигает значения, мало отличающегося от минимального.

Наконец, третий этап — уточнение решения (итерации от 36-й до 60-й). На этом этапе по-прежнему $x^i(T) \approx 0$, а понижение F_0

становится очень медленным, хотя управление может меняться заметно. На рис. 39 изображены управляющие функции $u_1(t)$ и $u_3(t)$ ($u_2(t) \equiv 0$, что соответствует и аналитическому решению [33], и приближенному в [41]), полученные на 63-й итерации. Показана также и функция $u_3(t)$ на 35-й итерации. Видно, что функция $u_3(t)$ имеет тенденцию превратиться в набор δ -функций, носители которых расположены вблизи точек $x^2(t) = 0$. Это видно из сравнения рис. 39 с рис. 40, на котором показаны функции $x^i(t)$,

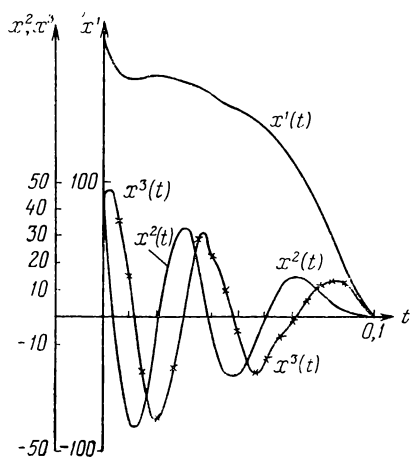


Рис. 40.

соответствующие управлению, полученному на 63-й итерации. Эти особенности численного решения помогут нам «угадать» структуру точного (предположим, что мы не знакомы с работой [33]).

2. Найденное в наших расчетах приближенное решение дает значение $F_0 = 2,515$, в то время как в [41] получено $F_0 = 3,55$, а «точное» значение $\min F_0 = 3,5$. То, что в наших расчетах $T = 0, 1$, а не 1, как в [41], только усугубляет ситуацию: ведь если найдено оптимальное решение на интервале $[0; 0,1]$, то оно продолжается на интервал $[0; 1]$ функциями $x^i(t) \equiv 0$, $u^i(t) \equiv 0$ с тем же

значением F_0 . Следовательно, при уменьшении T значение $\min F_0$ по крайней мере не убывает. Чтобы разобраться в противоречии, попробуем угадать точное решение. Возьмем, например, в качестве $u^i(t)$ δ -функции с полюсами в точке $t = 0$ и интенсивностями $-x^i(0)/a_i$, $i = 1, 2, 3$. Легко подсчитать значение F_0 для такого управления

$$F_0 = \frac{200}{100} + \frac{30}{25} + \frac{40}{100} = 3,6.$$

Однако, можно взять и δ -функции с полюсами в любой точке t^* и интенсивностями $-x^i(t^*)/a_i$. Так как $a_2 \ll a_1 = a_3$, естественно выбрать t^* так, чтобы $x^2(t^*) = 0$ (ближайшее к 0 такое $t^* \approx 0,0025$, затем они повторяются с периодом $\approx \pi/200 \approx 0,015$). В этом случае $F_0 = |x^1(t^*)/a_1| + |x^3(t^*)/a_3|$. Эта величина легко вычисляется с помощью интегралов (14): $F_0 = 2,5075$, что всего на 0,0075 меньше найденного нами численно значения $F_0 = 2,515$. Что касается расхождения с приведенным в [41], [86] теоре-

тическим значением $\min F_0 = 3,5$, то это связано с ошибкой в теоретической формуле, приведенной в [41], стр. 207. Правильная формула, видимо, выглядит так:

$$\min F_0 = \frac{1}{\alpha_1} \sqrt{\frac{\mu(\mu-\lambda)}{1-\lambda} [x^2(0)]^2 + [x^1(0)]^2} + \\ + \frac{\lambda}{\alpha_3} \sqrt{\frac{\mu(1-\mu)}{\lambda(1-\lambda)} [x^2(0)]^2 + [x^3(0)]^2}. \quad (15)$$

Здесь $\mu = 2/3$, $\lambda = 1/3$, $\alpha_1 = 100$, $\alpha_3 = 33^{1/3}$ — параметры, в терминах которых в [41] записана система уравнений движения (1). Это не что иное, как то решение, которое мы выше предположили оптимальным (δ -функции с полюсами в точках $x^2(t) = 0$). Если провести вычисления, получим $\min F_0 = 2,5075$. В [41] формула (15) приведена с ошибкой: пропущен множитель $\lambda = 1/3$ перед вторым радикалом, что и приводит к величине $\min F_0 = 3,5$ (кроме того, запись (15) в [41] содержит и две легко устранимые опечатки). Таким образом, методом локальных вариаций найдено решение с ошибкой в F_0 , превышающей 40%; в наших расчетах ошибка $\approx 0,4\%$.

3. Неединственность решения связана с тем, что в качестве носителя δ -функции можно взять любой из моментов $x^2(t) = 0$. Однако можно брать и функции, состоящие из нескольких δ -функций с полюсами в нулях $x^2(t)$, что и было получено численно (см. рис. 39). Видно также, что функция $u_1(t)$ допускает значительное отклонение от точной без существенного влияния на значение F_0 ; в $u_3(t)$ точная структура прослеживается довольно отчетливо.

4. Следующие причины побудили взять $T = 0,1$. Дело в том, что при $x^1 \approx 200$ функции $x^2, x^3(t)$ подобны $\sin 200t$. Поэтому численное интегрирование системы (1), претендующее на точность, скажем, 0,1%, требует в схеме второго порядка точности (4) шага $\Delta t \approx 0,001 - 0,0001$. (Это следует из несложной оценки точности разностной формулы (4)). В наших расчетах временная сетка для управления имела шаг $\Delta t = T/N = T/100$, однако интегрирование системы (1) осуществлялось меньшим шагом $dt = 0,1 \Delta t$, так что обеспечить необходимую точность интегрирования было бы не очень сложно и при $T = 1$. Но на интервале $[0, 1]$ функции $x^2, x^3(t)$ имели бы $\sim 50 - 60$ полувольт, а так как решение $x(t)$ запоминается в узлах сетки с шагом $\Delta t = T/N$, то мы имели бы около двух точек для описания полувольты. При интегрировании сопряженной системы решение $x(t)$ восстанавливается по имеющейся таблице $x(t_n)$ линейной интерполяцией, что при $T = 1$ приведет к заметным ошибкам. Конечно, можно (и не очень трудно) избежать и этой неприятности, если при интегрировании сопряженной системы восстанавливать необходимые значения $x(t)$

не линейной интерполяцией по полученной таблице, а одновременным интегрированием справа налево системы (1). Однако и в этом случае были бы получены довольно грубые численные решения, так как в используемом нами классе кусочно постоянных управлений шаг $\Delta t = 0,01$ был бы слишком грубым, выделяя положение точки $x^2(t) = 0$ с точностью почти до полуволны. В используемой в [41] расчетной схеме (4) для обеспечения точности следовало бы использовать сетку с 1000, по крайней мере, узлами (на интервале $[0; 1]$). В [41] нет данных о шаге сетки, но, судя по приведенным там рисункам, узлов было много меньше (можно предположить, что их было 64)*). Следовало бы проконтролировать полученные в [41] численные решения. Средства этого контроля хорошо известны и несложны: получив функции $u^i(t)$ в виде кусочно постоянных на сетке функций, нужно проинтегрировать задачу Коши (1) с этими $u^i(t)$, но с шагом, существенно меньшим шага сетки для u . Качество приближенного решения зависит от того, какие величины $x(T)$ будут при этом получены. Нами был проведен расчет, показавший, что шаг $\Delta t = 0,01$ слишком груб. Он состоял в том, что по функциям $x^i(t) = x^i(0)(1 - t/T)$ и формулам типа (4) были рассчитаны сеточные функции

$$u_{n+1/2}^1 = \frac{1}{a_1} \left\{ \frac{x_{n+1}^1 - x_n^1}{\Delta t} - A_1 \frac{x_n^2 + x_{n+1}^2}{2} \frac{x_n^3 + x_{n+1}^3}{2} \right\}, \quad x_n = x(t_n), \quad (16)$$

а затем с этими $u^i(t)$ была проинтегрирована система (1) с шагом $dt \ll \Delta t$.

Полученные функции существенно разошлись с положенными в основу расчета (16) линейными функциями $x^i(t)$.

Сравнение таблиц 1 и 2 показывает, что первая задача была решена намного быстрее второй, причем удачный выбор величин s^- , s^+ , ограничивающих допустимые размеры вариаций управления, является серьезным фактором, определяющим трудоемкость расчетов. Во второй задаче такого подбора s^- , s^+ не делалось; в табл. 2 представлены результаты, полученные, так сказать, «с первой попытки»: она оказалась удачной. Правда, до этого были неудачные попытки решить задачу на интервале времени $[0, 1]$. Неудачи были связаны с тем, что, рассчитав исходное управление по формуле (16), мы не получали $x^i(T) = 0$. Анализ возможных причин этого факта заставил внимательнее разобраться в качественном характере решений системы уравнений (8). В процессе этого анализа, возможно (точный ход событий сейчас восстановить уже нельзя), и были подобраны ограничения $|\delta u^1| \leq 1$; $|\delta u^2| \geq 0,5$; $|\delta u^3| \leq 0,5$. (Когда в расчетах что-то не получается,

*) Если это предположение верно, найденное в [41] управление не обеспечивает и выполнения условий $x(T) = 0$.

приходится проверять самые разные элементы алгоритма. Например, нет ли ошибки в программе, или, может быть, вариации управления столь велики, что нельзя пользоваться линейной теорией возмущений и т. д.) Специального подбора ограничений для $|\delta u^i|$ не производилось. Вероятно, это привело к какому-то перерасходу машинного времени. Однако, то, что первая же попытка решения задачи на интервале времени $[0; 0,1]$ оказалась успешной, сыграло и благоприятную роль: плохое совпадение предсказанных и фактических вариаций $x^2(T)$, $x^3(T)$, которое хорошо видно в табл. 2 (на первых итерациях), наводит на мысль об ошибке в программе.

Но более внимательный анализ (он приведен выше) показывает, что это несовпадение не так уж страшно, и даже естественно. Однако читатель должен понимать, что пренебрежение подобным расхождением оправдано прежде всего благополучным исходом расчета в целом. В дальнейшем решение второй задачи было повторено с единственным изменением: величины $s_{n+1/2}^-$, $s_{n+1/2}^+$, ограничивающие возможные значения переменных $\delta u_{n+1/2}$ в задаче линейного программирования, определяющей вариацию управления, не были фиксированы, как в расчетах, представленных в таблицах 1, 2. Был подключен алгоритм пересчета s^- , $s_{n+1/2}^+$. Вводилась последовательность $r_{n+1/2}^i$, $n=0,1,\dots,N-1$ (напомним, что $s_{n+1/2}$ суть вариации компонент управлений, т. е. $\delta v_{n+1/2}^i$, $\delta \omega_{n+1/2}^i$, $i=1, 2, 3$; N — число интервалов временной сетки для управления). После каждой итерации числа r пересчитывались по формуле

$$r_{n+1/2} := 0,8r_{n+1/2} + \text{sign } s_{n+1/2}.$$

Таким образом, r велико, если эволюция u носит монотонный характер. И наоборот, r мало, если u в процессе итераций или не меняется ($s_{n+1/2}=0$), или меняется немонотонно (то положительно, то отрицательно). На следующей итерации числа r используются при назначении s^- , s^+ . Например

$$s_{n+1/2}^+ = 0,2S + \xi |r_{n+1/2}|,$$

а параметр ξ подбирается так, чтобы среднее значение $s^+ = \frac{1}{N} \sum s_{n+1/2}^+$ имело заданную (такую же, как с постоянными $s_{n+1/2}^-$, $s_{n+1/2}^+$) величину. Результат решения задачи представлен в табл. 3, предсказанные значения $x^i(T)$ не приводятся — они примерно такие же, как и в табл. 2. Видно, что решение получено по крайней мере в два раза быстрее, причем решение терминальной задачи почти не убыстрилось (14 итераций вместо 18), а собственно оптимизация прошла намного успешнее. Это легко понять: ведь искомое решение вырождается в δ -функции, строить их вариациями, ограниченными постоянными «малыми» значениями $|\delta u^i| \leq 1$;

Т а б л и ц а 3

ν	F_0	$x^1 (T)$	$x^2 (T)$	$x^3 (T)$
0	0,25000	190,8	0,728	49,69
1	0,4520	169,37	38,87	25,56
2	0,5955	157,21	44,34	-1,81
3	0,9073	135,19	1,2252	-40,80
4	1,195	105,43	-37,08	-19,41
5	1,570	76,86	-19,46	35,42
6	1,8860	44,22	34,04	21,04
7	2,0566	33,62	35,09	-0,26
8	2,3340	16,15	28,24	-8,07
9	2,7292	-8,29	-1,67	-24,53
10	2,5443	-0,038	0,14	-25,75
11	2,6295	0,902	-2,04	-19,86
12	2,7216	1,04	-4,43	-13,50
13	2,8462	0,24	-3,0	-7,21
14	2,9835	-0,70	0,22	0,07
15	2,8826	0,016	0,33	-1,46
16	2,8319	-0,0012	0,060	-0,092
17	2,7956	-0,0062	0,025	-0,058
18	2,7575	0,0026	-0,056	-0,051
19	2,7208	0,0044	0,058	0,11
20	2,6766	0,0076	-0,056	-0,21
22	2,5999	-0,0074	0,048	-0,0097
24	2,5366	-0,0074	0,014	-0,049
25	2,5197	0	0,029	-0,075
26	2,5185	0,0006	0,084	-0,030
27	2,5170	0,020	-0,11	-0,085
28	2,5166	0,0094	0,18	-0,064
29	2,5152	0,0044	-0,057	-0,16
30	2,5152	0,013	0,15	0,051
31	2,5137	0,027	0,07	0,11

$|\delta u^3| \leq 0,5$, очень долго (на рис. 39 $\max_t |u_3(t)| \approx 30$). Описанный выше прием приводит к эффективному увеличению шага вариаций на тех интервалах сетки, на которых сосредоточено управление. Именно с этим и связано ускорение расчета. Полученная в этом расчете функция $u^3(t)$ (на 30-й итерации) имеет тот же характер, что и $u^3(t)$ на рис. 39, однако сходство с δ -функциями стало еще более резким: $\max_t |u^3(t)| \approx 45-50$, соответственно уменьшились

и размеры «носителя» $u^3(t)$; каждый из четырех первых «пиков» в u^3 расположен на 2-х—3-х счетных интервалах, т. е. на отрезках времени $\approx 0,002-0,003$. Сузился и носитель последнего пика в u^3 . Заметим, что последний расчет был проведен не методом проекции градиента (когда вариация управления находится решением задачи квадратичного программирования), а методом последовательной линеаризации (вариация находится решением задачи линейного программирования).

§ 35. Модельная задача с фазовым ограничением и разрывом фазовой траектории

Следующая простая задача предложена А. А. Милютиним в качестве теста для испытаний приближенных методов.

Найти управление $u(t)$, $0 \leq t \leq 2$, минимизирующее функционал

$$F_0[u(\cdot)] = \int_0^2 x^2(t) dt,$$

определенный на решении системы

$$dx/dt = u(t), \quad x(0) = 0$$

при дополнительных условиях:

- 1) $x(2) = 0 \quad (F_1[u(\cdot)] = x(2));$
- 2) $x(t) \geq 1 \quad \text{при } 0,5 \leq t \leq 1,5$

или $F_2[u(\cdot)] \geq 0$, где

$$F_2[u(\cdot)] = \min_{[0,5; 1,5]} \{x(t) - 1\}.$$

Точное решение этой задачи очевидно:

$$u(t) = \delta(t - 0,5) - \delta(t - 1,5),$$

$$x(t) = \begin{cases} 0 & \text{при } t \notin [0,5; 1,5], \\ 1 & \text{при } t \in [0,5; 1,5]. \end{cases}$$

Численное решение ее связано с преодолением следующих трудностей:

1) необходимо получить решение, удовлетворяющее фазовому ограничению $x(t) \geq 1$ на $[0,5; 1,5]$, нарушить которое очень выгодно с точки зрения минимизации функционала F_0 ;

2) необходимо построить численные аналоги δ -функций в управлении $u(t)$.

Эту задачу автор решал итерационным методом, подробно описанным в §§ 19—21. Коротко опишем структуру одного шага процесса.

Пусть имеется некоторое управление $u(t)$.

1. Интегрируя уравнение $\dot{x} = u$, $x(0) = 0$, находим $x(t)$ и вычисляем значения функционалов F_0 , F_1 , F_2 .

2. Вычисляется функциональная производная

$$\frac{\partial F_0[u(\cdot)]}{\partial u(\cdot)} = w_0(t),$$

для чего интегрируется задача Коши

$$-\dot{\psi} = 2x(t), \quad \psi(2) = 0 \quad (w_0(t) = \psi(t)).$$

3. Интервал фазового ограничения $[0,5; 1,5]$ разбивается на k равных частей, на каждой из них находится точка минимума $x(t)$, t^j , $j=1, 2, \dots, k$.

4. Формируется и решается задача определения вариации управления $\delta u(t)$:

$$\min_{\delta u(\cdot)} \int_0^2 w_0(t) \delta u(t) dt$$

при условиях

$$a) \quad x(2) = \int_0^2 \delta u(t) dt = 0;$$

$$b) \quad x(t^j) + \int_0^t \delta u(t) dt \geq 1, \quad j=1, 2, \dots, k;$$

$$c) \quad s^-(t) \leq \delta u(t) \leq s^+(t),$$

где s^- , $s^+(t)$ — некоторые малые функции, определяющие шаг процесса и обеспечивающие достаточную точность линеаризации исходной задачи. В процессе решения функции $s^-(t)$, $s^+(t)$ претерпевали определенную эволюцию, имеющую целью допустить вариацию δu большей величины на тех участках интервала времени, где она «более полезна» (см. § 34). Алгоритм был реализован на равномерной сетке с 100 узлами; функции $u(t)$, $\delta u(t)$, s^- , $s^+(t)$ были кусочно-постоянными; например, $u(t) = u_{n+1/2}$, при $t \in (t_n, t_{n+1})$, $t_n = n\tau$, $\tau = 0,02$. Для функций такого класса задача определения $\delta u(t)$ становится задачей линейного программирования размером $(k+2) \times 100$.

Функционал $F_0[u(\cdot)]$ вычислялся точным интегрированием в классе кусочно-линейных функций $x(t)$; интегрирование уравнения $-\dot{\psi} = 2x(t)$ также было точным в классе кусочнолинейных $x(t)$. Эти предосторожности были связаны с наличием особенностей в искомом решении; использование более простых аппроксимаций, которые были бы вполне приемлемы на гладких траекториях $\{u(\cdot), x(\cdot)\}$, в данном случае приводит к заметным ошибкам (см. в связи с этим также стр. 224).

В табл. 1 показан ход итерационного процесса (при $k=3$ и при $k=5$) эволюцией следующих величин: v — номер итерации, значение $F_0[u^v(\cdot)]$, значение $F_1[u^v(\cdot)] = x(2)$, $F_2[u^v(\cdot)] \equiv$

$$\equiv \min_{[0,5; 1,5]} x(t), \quad F_0^* = \int_{0,5}^{1,5} x^2(t) dt. \text{ Последняя величина характеризует}$$

суммарное нарушение фазового ограничения. Задача линейного программирования решалась итерационным процессом, описанным в § 48.

Т а б л и ц а 1

Расчет с $k=3$						Расчет с $k=5$					
ν	F_0	$x(2)$	F_2	F_0^*	i	ν	F_0	$x(2)$	F_2	F_0^*	i
0	0,0515	0,196	+0,100	0,031	0	0	0,05	0,20	0,10	0,03	0
1	0,186	0,268	0,200	0,128	0	1	0,30	0,60	0,20	0,147	0
2	0,432	0,400	0,300	0,291	0	2	0,61	0,72	0,30	0,337	0
3	0,583	0,350	0,400	0,428	0	3	1,02	0,84	0,40	0,606	0
5	0,986	0,250	0,600	0,791	0	5	1,16	0,60	0,60	0,823	0
7	1,438	0,150	0,800	1,087	0	7	1,42	0,38	0,80	1,108	0
9	1,681	0,065	0,965	1,048	3	11	1,65	0,30	1,17	2,111	1
11	1,384	-0,001	0,993	1,079	1	13	1,66	-0,00	1,00	1,348	2
15	1,202	-0,001	0,943	0,985	2	15	1,27	0,00	1,00	1,019	1
20	1,125	0	0,930	0,987	2	17	1,17	0,00	0,95	0,978	10
25	1,095	0,007	0,977	1,079	2	20	1,14	0,00	0,97	0,987	1
30	1,071	-0,006	0,977	0,995	2	23	1,10	0,00	0,97	0,992	3
35	1,061	-0,003	0,985	1,040	1	26	1,08	0,00	0,95	0,998	10
40	1,037	0,007	0,975	0,993	4	29	1,06	0,00	0,97	0,997	4
45	1,033	-0,001	0,968	0,995	4	33	1,04	0,00	0,96	0,993	10
50	1,041	0	0,979	1,03	1	37	1,03	0,00	0,97	0,998	9
55	1,037	0	0,970	1,03	1	40	1,04	0,00	0,98	1,009	9
65	1,031	0	0,992	1,02	3	43	1,03	0,00	0,98	1,001	7
70	1,024	-0,002	0,985	0,998	5	45	1,02	0,00	0,96	0,994	3

В таблице приведено i — число итераций (пересчетов вектора g в этом алгоритме).

Рис. 41 иллюстрирует процесс поиска решения — на нем изображены функции $x(t)$ на нулевой 9-й, 25-й и 70-й итерациях.

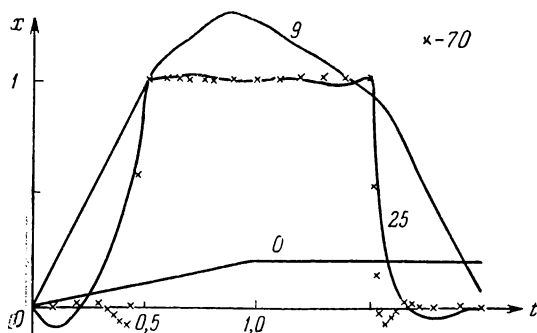


Рис. 41.

Отметим характерные черты процесса поиска и численного решения.

1. На первых итерациях решается «терминальная» задача — находится траектория $\{u(\cdot), x(\cdot)\}$, удовлетворяющая всем допол-

нительным условиям задачи. Процесс минимизации начинается с 10-й итерации.

2. Функция $x(t)$ довольно быстро стабилизируется в зоне фазового ограничения и вне окрестностей полюсов δ -функций в $u(t)$. Большое число итераций связано с построением δ -функций суммированием малых возмущений $\delta u(t)$. При этом постепенно уменьшается ширина зоны «размазывания» δ -функций, а значение функционала F_0 понижается незначительно.

3. Эффективность построения δ -функций зависит от величины разрешаемой на каждом счетном интервале вариации δu :

$$s_{n+1/2}^- \leq \delta u_{n+1/2} \leq s_{n+1/2}^+.$$

Как уже отмечалось, числа s^- , $s_{n+1/2}^+$ определенным образом менялись в ходе поиска. Именно, на каждом счетном интервале была определена величина $r_{n+1/2}$, которая после каждого шага пересчитывалась и использовалась для назначения s^- , s^+ так, как это описано в § 34.

4. На рис. 41 видно, что вне зоны фазового ограничения, где точное решение есть $x(t)=0$, численное колеблется около нуля. Это можно объяснить двумя факторами. Во-первых, такие отклонения от точного решения дают малый вклад в $\int x^2 dt$, и их трудно убрать процессом вариаций первого порядка, особенно при наличии мощного «конкурента» — нарушения фазового ограничения. Вторая причина не столь очевидна и связана с характером точного решения задачи в классе кусочно линейных функций.

Рассмотрим часть нашей задачи

$$\min \int_0^1 x^2(t) dt, \quad x(0)=0, \quad x(1)=1$$

в классе кусочно линейных функций, когда $x(t)$ определяется значениями $0=x_0, x_1, x_2, \dots, x_N=1$ и

$$x(t) = x_n + \frac{1}{\tau}(t - n\tau)(x_{n+1} - x_n) \quad \text{при} \quad t_n \leq t \leq t_{n+1}, \quad t_n = n\tau.$$

Напрашивающаяся аппроксимация очевидного точного решения кусочно линейной функцией с значениями $x_0=x_1=\dots=x_{N-1}=0, x_N=1$ — неоптимальна. Оптимальная функция определяется решением разностного уравнения (аналог уравнения Эйлера) *)

$$x_{n-1} + 4x_n + x_{n+1} = 0; \quad x_0 = 0; \quad x_N = 1.$$

*) Оно легко получается варьированием формулы

$$\int_0^1 x^2(t) dt = \sum_{n=0}^{N-1} \frac{\tau}{3} (x_n^2 + x_n x_{n+1} + x_{n+1}^2).$$

Решение: $\dots, x_{N-3} \approx -0,02; x_{N-2} \approx +0,08; x_{N-1} \approx -0,27; x_N = 1$. Оно колеблется, амплитуда колебаний убывает примерно в $(1 + \sqrt{3})$ раз за шаг. Численное решение имеет такой же качественный характер, но, так сказать, с эффективным шагом, в несколько раз большим действительного.

5. Точное значение $F_0 = 1$; приближенный метод дал решение с $F_0 \approx 1,025$. Ошибка в 2,5% состоит из двух частей. Первая часть — это ошибка аппроксимации, возникшая из-за сужения задачи на класс кусочно линейных функций $x(t)$. Эта ошибка имеет порядок шага τ сетки $\{t_n\}$ и может быть вычислена по указанному выше точному решению задачи в классе кусочно линейных $x(t)$. При шаге сетки $\tau = 0,02$ точное сеточное решение дает $F_0 = 1,0130$ (для «направляющейся» аппроксимации $F_0 = 1,0133$).

Вторая часть ошибки $\approx 1,2\%$ — это ошибка поиска, связанная, в частности, с «размазанностью» δ -функций. Ошибку аппроксимации легко уменьшить, уменьшив τ ; на ошибку поиска это практически не повлияет, ее уменьшение может быть достигнуто увеличением числа итераций, уменьшением S и, быть может, увеличением точности некоторых промежуточных вычислений. Поэтому «размазанность» δ -функции следует характеризовать не числом счетных интервалов, а шириной временного интервала: именно она сохранится, если расчеты повторить, изменив лишь шаг сетки.

6. Точность соблюдения фазового ограничения может быть легко улучшена, если процесс поиска завершить несколькими итерациями с значительно меньшим шагом S (см. теорему 1, § 24).

7. Сравнивая решение при $k=3$ с решением при $k=5$, видим, что та же точность получена за меньшее число итераций. Увеличение числа точек k , аппроксимирующих фазовое ограничение, позволило использовать больший шаг, и поиск протекал быстрее. Но зато сама итерация стала более трудоемкой, и поэтому временная цена обоих решений примерно одинакова.

По этой же программе решалась несколько иная модельная задача, осложненная еще ограничением на u : найти

$$\min \int_0^2 x^2(t) dt$$

на решении уравнения

$$\dot{x} = u; \quad x(0) = 1; \quad u \geq 0$$

при условиях

$$x(2) = 3,5;$$

$$x(t) \geq 1 + \sqrt{2t} \quad \text{при } t \in [0,5; 1,5].$$

Решение задачи очевидно:

$$x(t) = \begin{cases} 1 & \text{при } 0 \leq t < 0,5, \\ 1 + \sqrt{2t} & \text{при } 0,5 \leq t < 1,5, \\ 1 + \sqrt{3} & \text{при } 1,5 \leq t < 2, \\ 3,5 & \text{при } t = 2. \end{cases}$$

Характер поиска решения показан на рис. 42 функциями $x(t)$

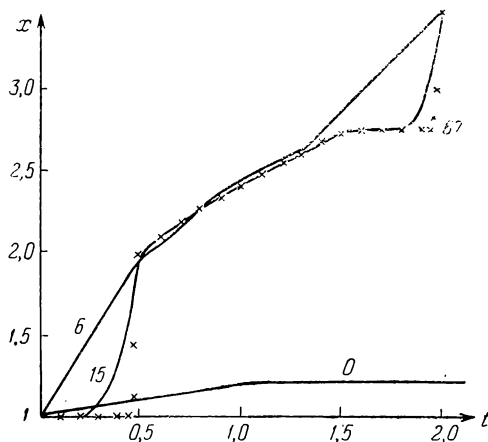


Рис. 42.

на нулевой, 6-й, 15-й и 67-й итерациях. Точное значение функционала $F_0 = 10,02$, приближенное — $F_0 = 10,11$; при этом фазовое ограничение выполнено с точностью $x(t) > 1 + \sqrt{2t} - 0,01$,

Таблица 2

v	t									
	0.36	0.38	0.40	0.42	0.44	0.46	0.48	0.5	0.52	
0	0,200	0,200	0,200	0,200	0,200	0,200	0,200	0,200	0,200	
10	2,71	3,53	3,53	3,53	4,02	4,03	4,02	4,36	1,21	1,06
15	2,45	3,90	4,85	4,85	5,95	5,95	6,39	6,86	0,94	1,09
20	1,34	1,27	3,27	4,39	7,98	7,98	8,59	10,51	1,37	1,29
25	0,69	0,01	1,35	2,69	6,25	10,16	12,53	14,71	1,26	1,34
30	0,03	0,03	0,03	0,19	5,14	10,17	14,04	18,28	1,36	1,40
35	0	0	0	0	4,64	9,62	15,27	20,47	1,14	1,16
40	0,16	0,13	0,12	0,06	3,66	8,90	14,80	21,79	0,92	0,94
45	0	0	0	0	2,67	7,91	15,92	23,51	1,00	1,00
50	0	0	0	0	1,77	7,13	16,21	24,87	0,86	0,87
55	0,01	0,01	0,01	0,01	1,32	6,21	16,33	26,06	0,92	1,08
60	0	0	0	0	1,04	5,87	16,13	26,53	1,13	0,99
67	0	0	0	0	0,71	5,79	15,93	27,04	1,14	0,98

$x(2) = 3,5 + 0,001$, в точном решении $\int x^2 dt = 5,79$, в приближенном — 5,82. Общая ошибка в значении $F_0^{1,5} \sim 0,09$ есть сумма ошибки аппроксимации $\sim 0,07$ и ошибки поиска $\sim 0,03$. Табл. 2 иллюстрирует процесс построения δ -функции; на ней приведены последовательные значения $u_{n+1/2}$ на интервалах, примыкающих к полюсу δ -функции в точке $t = 0,5$ (v — номер итерации).

§ 36. Оптимальный режим остановки реактора

Уравнения «движения» управляемой системы просты:

$$\left. \begin{aligned} \frac{dx^1}{dt} &= Ax^2 + Bu - Cx^1 - Dx^1u, \\ \frac{dx^2}{dt} &= -A'x^2 + A'u, \end{aligned} \right\} \quad (1)$$

$$0 \leq t \leq T.$$

Начальные условия

$$\Gamma(x) = 0: \quad x^1(0) = 1; \quad x^2(0) = 1. \quad (2)$$

На искомое управление $u(t)$ наложено ограничение

$$u(t) \in U: \quad 0 \leq u \leq 1. \quad (3)$$

Процесс рассматривается на интервале времени $[0, T^*]$, где $T^* > T$. При этом предполагается $u(t) \equiv 0$ при $t \in [T, T^*]$. Задача состоит в выборе функции $u(t)$, минимизирующей функционал

$$F_0[u(\cdot)] \equiv \max_{0 \leq t \leq T^*} x^1(t). \quad (4)$$

Таким образом, мы имеем задачу на отыскание

$$\min_{u(\cdot)} \max_t x^1(t). \quad (4^*)$$

Физическое содержание задачи. Уравнения (1) описывают средние значения концентраций радиоактивных ксенона (x^1) и йода (x^2) в ядерном реакторе, причем используется простейшая «точечная» математическая модель. В действительности x^1 и x^2 — суть функции не только времени, но и трех пространственных координат, а уравнения (1) в более точной постановке задачи были бы заменены существенно более сложной системой уравнений с частными производными. Функция $u(t)$ есть среднее значение потока нейтронов в реакторе. Это значение поддается регулированию и в данной постановке задачи играет роль управления. Ограничение $u(t) \geq 0$ имеет очевидный физический смысл, ограничение $u(t) \leq 1$ связано с техническими возможностями аппарата. A, B, C, D, A' — некоторые заданные постоянные

числа, зависящие от технических данных реактора. Задача сформулирована в безразмерном виде. Состояние $\{x^1=1, x^2=1; u=1\}$ является стационарным для (1), так как при этом $\dot{x}^1=\dot{x}^2=0$. Необходимо остановить реактор, т. е. перейти в состояние с $u(t) \equiv 0$, T есть заданное время остановки (время переходного процесса). Ищется такой режим $u(t)$, при котором минимально значение «отравления» реактора ксеноном (4). Дело в том, что режим остановки должен удовлетворять следующему условию: начав выключение реактора в момент $t=0$, нужно иметь возможность

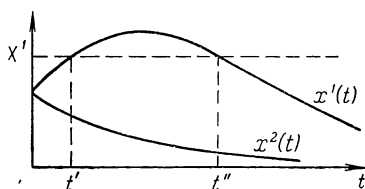


Рис. 43.

в любой момент времени $t > 0$, если возникнет внезапная необходимость вновь включить его, т. е. вывести на стационарный уровень. Если, например, осуществить самый простой режим остановки $\{u(t) \equiv 0 \text{ при } t_i > 0\}$, то эволюция концентраций ксенона и йода в соответствии с уравнениями (1) будет происходить так,

как это показано на рис. 43 (качественно). Повышение x^1 (так называемое «ксеноновое отравление») приводит к изменению важной характеристики реактора («реактивности»). В частности, при $x^1(t) > X^1$, где X^1 определяется техническими данными аппарата («запасом реактивности»), реактор становится неуправляемым, в нем возможен только нерабочий режим $u(t) \equiv 0$. Таким образом, в этом случае, если необходимость включения реактора возникнет на интервале времени (t', t'') (см. рис. 43), этого сделать не удастся. Величина интервала $(t'' - t')$ для разных типов реакторов может варьироваться от нескольких часов до суток, так что пренебречь ею нельзя, особенно в аппаратах специального назначения. Физическая постановка задачи включает в себя еще одно ограничение технического характера, связанное с реальными возможностями изменения $u(t)$:

$$|du/dt| \leq \alpha u. \quad (5)$$

Однако величина α^{-1} настолько мала относительно характерных времен в данной задаче ($\alpha T \sim 10^3 - 10^4$), что ограничением (5) можно совсем не пользоваться и принять для $u(t)$ модель произвольной («измеримой») функции. Если найденное при такой идеализации оптимальное решение $u(t)$ окажется разрывным, а число разрывов будет невелико (именно так и окажется), то аппроксимация разрывных решений, даже обращающихся в нуль, функцией, удовлетворяющей условию (5), особых трудностей не представляет, а ошибка такой аппроксимации (относительно значения функционала F_0) очень мала и заведомо меньше неточности самой модели (1).

Решение задачи методом последовательной линейризации (§§ 19—21). Задача (1)—(4) была одной из первых задач оптимального управления, решавшейся автором (в 1963 г.). Многие детали технологии тогда еще только отрабатывались, и сейчас можно указать на некоторые неудачные решения. Несмотря на это, задача была успешно решена, проведена большая серия расчетов и составлена таблица оптимальных управлений и минимальных значений F_0 . В этой таблице было два аргумента: время управления T и мощность реактора z , через которую вычисляются коэффициенты A, B, C, D системы уравнений. Анализ этого множества оптимальных решений позволил угадать гипотетическую структуру точного решения задачи. К этому вопросу мы еще вернемся, а сейчас опишем метод решения задачи в той форме, в которой он был реализован в 1963 г. Прежде всего, математическая формулировка задачи была изменена: $u(t)$ было отождествлено с дополнительной фазовой переменной $x^3(t)$, а в качестве произвольного управления использовалась функция $v(t) = \dot{u}$. Задача сразу же усложнилась. Система уравнений приняла вид]

$$\begin{aligned}\frac{dx^1}{dt} &= Ax^2 + Bx^3 - Cx^1 - Dx^1x^3; & x^1(0) &= 1; \\ \frac{dx^2}{dt} &= -Ax^2 + Ax^3, & x^2(0) &= 1; \\ \frac{dx^3}{dt} &= v, & x^3(0) &= 1, \\ 0 &\leq t \leq T.\end{aligned}\tag{6}$$

Минимизируется функционал

$$F_0[v(\cdot)] \equiv \max_t x^1(t).\tag{7}$$

Вместо простого условия $0 \leq u(t) \leq 1$ появляются гораздо более сложные ограничения в фазовом пространстве:

$$\begin{aligned}F_1[v(\cdot)] &\equiv \max_{0 \leq t \leq T} x^3(t) \leq 1, \\ F_2[v(\cdot)] &\equiv \min_{0 \leq t \leq T} x^3(t) \geq 0,\end{aligned}\tag{8}$$

и условие на правом конце траектории

$$F_3[v(\cdot)] \equiv x^3(T) = 0.\tag{9}$$

Заметим, что с бесконечным интервалом времени в (7) никаких особых трудностей нет. Можно либо указать такое значение T^* (не очень большое, $T^* \sim 2T$), что $\max_{0 \leq t \leq \infty} x^1(t) = \max_{0 \leq t \leq T^*} x^1(t)$, либо,

решив аналитически систему (1*) с $u(t) = 0$ на интервале $[T, \infty]$ и найдя $\max_{T \leq t} x^1(t) = \Phi[x^1(T), x^2(T)]$, заменить выражение (7) на выражение

$$\max \{ \max_{0 \leq t \leq T} x^1(t), \Phi[x^1(T), x^2(T)] \}. \quad (10)$$

В расчетах использовался первый способ (назначение T^*). Аналитическое выражение для Φ мы не выписываем, так как оно без труда может быть получено читателем.

Итак, вместо задачи (1)–(4), в которой был лишь один функционал F_0 , не имеющий производной Фреше, мы получили задачу с тремя функционалами, F_0, F_1, F_2 , дифференцируемыми лишь по направлениям, и с одним, дифференцируемым по Фреше, функционалом F_3 . Правда, в новой задаче нет геометрических ограничений на управление, но учет таких ограничений менее всего затруднителен в расчетах. Однако это усложнение было оправдано, объяснить причины удобнее несколько позже. Задача решалась по схеме §§ 19–21. Сетка для управления состояла из 64 точек, причем из них 50 приходилось на «активный участок» $[0, T]$, остальные — на пассивный (T, T^*) . Интегрирование самой системы (6) осуществлялось с шагом, заметно меньшим шага сетки для управления. Вариация функционала F_0 аппроксимировалась тремя точками τ_i , для аппроксимации F_1 и F_2 использовалось по две точки. Нужно иметь в виду, что отсутствие производных Фреше у F_0, F_1, F_2 есть существенное обстоятельство, так как оптимальная и близкие к ней траектории имеют следующую структуру: определим на $[0, T^*]$ множества:

$$M_0 \equiv \{t: x^1(t) \geq \max_t x^1(t) - \varepsilon\}, \quad \varepsilon > 0,$$

$$M_1 \equiv \{t: x^3(t) \geq 1 - \varepsilon\},$$

$$M_2 \equiv \{t: x^3(t) < \varepsilon\}.$$

Множество M_0 состоит из интервала длиной μ_0 и одной точки $t^* > T$, M_1 и M_2 — суть интервалы (на $[0, T]$) длиной μ_1 и μ_2 . При этом $\mu_0 + \mu_1 + \mu_2 = T$. Таким образом, задача определения вариации управления $\delta v(\cdot)$ приводила к задаче линейного программирования (см. § 48) с числом неизвестных $N=50$, с числом строк равным 8: три строки аппроксимировали δF_0 , две — δF_1 , две — δF_2 и одна — δF_3 ; вектор $e = \{1; 1; 1; 0; 0; 0; 0; 0\}$. Реализация одной итерации (вычисление $\delta v(t)$ и переход от $v(t)$ к $v(t) + \delta v(t)$) требует следующих вычислений:

1. Интегрирование системы (6) и вычисление F_0, \dots, F_3 .

2. Трехкратное интегрирование сопряженной системы с начальными данными $\psi = \{1; 0; 0\}$, заданными в точках аппроксимации δF_0 : τ_1, τ_2, τ_3 ; интегрирование ведется справа налево; при $t > \tau_i$ полагаем $\psi^{(i)}(t) \equiv 0$.

3. Интегрирование сопряженной системы для точек аппроксимации τ_4, τ_5 (для δF_1), τ_6, τ_7 (для δF_2) и $\tau_8 = T$ (для δF_3) с начальными данными $\psi(\tau_i) = \{0; 0; 1\}$ времени фактически не занимает, так как

$$\psi(t) = \begin{cases} \{0; 0; 1\} & \text{при } t \leq \tau_i, \\ \{0; 0; 0\} & \text{при } t > \tau_i. \end{cases}$$

4. Решение задачи линейного программирования. В качестве начального управления задавалась обычно функция

$$u(t) = (1 - t/T); \quad v(t) = -1/T, \quad (11)$$

хотя некоторые расчеты для контроля проводились и при других исходных данных. Стандартный расчет состоял из 50—60 итераций (30 минут на машине М-20, имеющей 4096 ячеек оперативной памяти и быстродействие $2 \cdot 10^4$ операций в секунду). Последние 10—20 итераций не давали, по существу, понижения значения F_0 . Как оказалось в дальнейшем, значение $\min F_0$ находилось с точностью 1%—0,5%. Это выяснилось после того, как были получены точные решения задачи, угаданные в результате анализа численных.

Расчеты показали, что оптимальная траектория может иметь две следующие характерные формы.

1. При малых временах управления $T \leq T_{кр}(z)$ (напомним, что z — параметр системы, определяющий значения A, B, C, D) оптимальное $u(t)$ имеет вид

$$u(t) = \begin{cases} 0 & \text{при } 0 \leq t < t_1, \\ 1 & \text{при } t_1 < t \leq T; \end{cases} \quad (12)$$

t_1 , разумеется, есть функция от z и T ; соответствующие кривые $T_{кр}(z)$, $t_1(z, T)$ были построены по результатам расчетов и частично приведены в работе [2]. Функция $x^1(T)$ в этом случае имела два локальных максимума: в точке $t_1 < T$ и в точке $t' > T$, причем $x^1(t') > x^1(t_1)$ и, следовательно, функционал F_0 является дифференцируемым по Фреше, так как множество M_0 состоит только из одной точки t' . По мере приближения T к $T_{кр}(z)$ значения $x^1(t_1)$ и $x^1(t')$ выравниваются, и при $T > T_{кр}$ оптимальный режим (12) недействителен, он заменяется другим, более сложным. Заметим, что $T_{кр}$ сравнительно мало, и режим (12) практически не очень интересен, так как он дает слишком большие значения $\min F_0$.

2. При $T > T_{кр}$ оптимальное $u(t)$ имеет вид

$$u(t) = \begin{cases} 0 & \text{при } 0 \leq t < t_1 \quad (t \in M_2), \\ u^*(t) & \text{при } t_1 \leq t < t_2 \quad (t \in M_0), \\ 1 & \text{при } t_2 \leq t \leq T \quad (t \in M_1). \end{cases} \quad (13)$$

Функция $x^1(t)$ устроена следующим образом: на интервале $(0, t_1)$ она возрастает до значения $F_0 = \max_t x^1(t)$, на интервале (t_1, t_2) она остается постоянной, $x^1(t) = F_0$, затем на интервале (t_2, T) падает, а на пассивном участке (T, T^*) сначала возрастает, в точке τ_3 достигает максимума, равного F_0 , затем спадает (асимптотически, при $t \rightarrow \infty$, до нуля). На рисунке 44 изображено полученное численно оптимальное решение $\{x^1(t), x^3(t)\}$. Отчетливо просматривается структура (13), хотя функция $u(t) = x^3(t)$, конечно, далеко не идеальна.

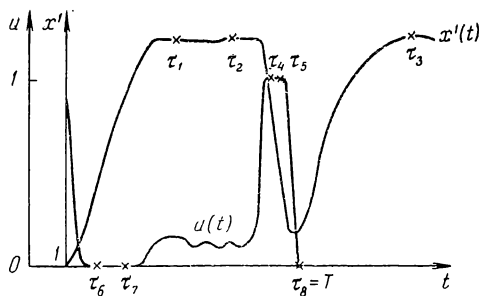


Рис. 44.

Точное решение задачи можно найти, зная структуру (13). Оно определяется двумя параметрами t_1, t_2 , вычисление которых требует решения некоторой нелинейной системы уравнений. Мы ограничимся здесь только общими указаниями, не доводя дела до окончательных формул. Итак, пусть заданы числа t_1, t_2 .

1. На $(0, t_1)$ интегрируется система (1) с $u(t) = 0$. Получаем значение $F_0 = x^1(t_1)$.

2. На (t_1, t_2) условия $x^1(t) = F_0$ и $\dot{x}^1 = 0$ дают возможность выразить $u^*(t)$ из первого уравнения (1):

$$u^*(t) = \frac{Ax^2(t) - CF_0}{B + DF_0} = \alpha x^2(t) + \beta. \quad (14)$$

Подставляя (14) во второе уравнение (1), получим уравнение

$$\dot{x}^2 = (\alpha - A)x^2 + A\beta, \quad (15)$$

которое элементарно интегрируется на (t_1, t_2) с начальными данными $x^2(t_1)$, полученными выше.

3. На интервале (t_2, T) система (1) интегрируется с $u(t) = 1$ и с известными данными Коши $x(t_2)$.

4. На (T, ∞) система (1) интегрируется с $u(t) = 0$, и определяется значение $x^* = \max_{t > T} x^1(t)$. Заметим, что все вычисления могут быть выполнены в конечном виде.

Описанная выше процедура определяет величины F_0, x^* как функции t_1, t_2 . После этого можно обычными методами решить задачу: найти

$$\min_{t_1, t_2} F_0(t_1, t_2) \text{ при условии } F_0(t_1, t_2) = x^*(t_1, t_2).$$

В наших расчетах (облегчавшихся наличием хороших начальных приближений для t_1, t_2) сначала при фиксированном t_1 находилось $t_2(t_1)$ решением (методом Ньютона) уравнения $F_0(t_1, t_2) = x^*(t_1, t_2)$. После этого F_0 становится функцией только одного параметра t_1 ; то, что эта функция определена некоторым алгоритмом, а не формулами, не очень важно. Затем метод параболической аппроксимации позволял без особого труда найти $\min_{t_1} F_0$.

Мы не будем вдаваться в детали вычислительной технологии, так как они в основном аналогичны описанным в § 20, хотя, конечно, применялись в то время в менее четкой форме. Особенно следует отметить важность нормировки функционалов.

Отметим теперь те методически неудачные моменты в построении алгоритма численного решения задачи, о которых уже упоминалось, и которые, безусловно, помешали получить результаты с меньшими затратами машинного времени.

1. Следовало $x^3(0)$ считать не фиксированным значением, а элементом управления, и искать его одновременно с $v(t)$. Фиксируя $x^3(0)=1$, мы ввели в $u(t)$ еще одну точку разрыва при $t=0$, а ведь в терминах $v(t)$ получить разрыв в $u(t)$ — это значит построить δ -функцию в $v(t)$ процессом малых вариаций $v(t) \rightarrow v(t) + \delta v(t)$. А это всегда трудно и приводит к большому числу итераций и «размазыванию» разрыва в $u(t)$.

2. Следовало отбросить условие $x^3(T)=0$, заменив систему (1) на $(0, T^*)$ составной системой, что соответствует задаче только на $(0, T)$ с заменой выражения (4) для F_0 на выражение (10). Это привело бы к устранению еще одной δ -функции в $v(t)$, имеющей полюс в точке T .

Отбросить условие $x^3(T)=0$ тем более выгодно, что его нужно было получать с высокой степенью точности: $|x^3(T)| \leq 10^{-6}$; в противном случае в некоторых вариантах (при больших мощностях z) отличие $u(t)$ от нуля на (T, T^*) заметно влияло на значение

$$\max_{t > T} x^1(t).$$

3. Наконец, был использован не самый удачный способ размещения точек аппроксимации на множествах M_0, M_1, M_2 , а именно: в качестве точек τ_1, τ_2, τ_3 выбирались узлы сетки t_n для управления, в которых значения $x^1(t_n)$ больше, чем во всех остальных. Тем не менее задачи решались надежно, с хорошей точностью и за вполне разумное машинное время. С учетом накопленного с тех пор опыта можно было бы сократить число итераций. Обсудим целесообразность замены управления $\dot{y}=v$, которая, как отмечалось, существенно усложнила форму задачи. Когда автор начинал эту работу, не имея еще опыта решения подобных задач, замена была произведена с тем, чтобы иметь дело с более гладкими функциями $\dot{u}(t) = \delta x^3(t)$, удовлетворяющими уравнению $\delta \dot{x}^3 = \delta v$

с малой правой частью. В свою очередь это должно было привести к лучшей точности в аппроксимации

$$\delta F_0 [\delta v(\cdot)] \approx \max_{i=1, 2, 3} \delta x^1(\tau_i). \quad (16)$$

Однако в дальнейшем, когда были сделаны попытки обойтись без замены u на v и решать задачу в более простой (формально) постановке (1)–(4), обнаружились и более глубокие причины, оправдывающие переход к задаче (6)–(9).

Это особенно ярко проявляется в задачах для аппаратов с большой мощностью z . Дело в том, что в этом случае коэффициенты

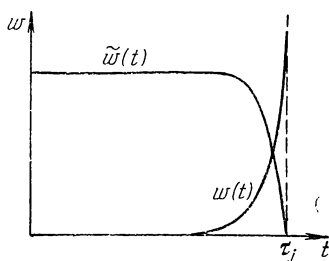


Рис. 45.

A, B, C, D очень велики (≈ 100) при $T \approx 1,0 \div 2,0$. Заметим, что с этим связаны и определенные трудности интегрирования системы (6) и сопряженных систем. Грубо говоря, шаг численного интегрирования dt должен быть таким, чтобы выполнялось неравенство $|dt\Lambda| \ll 1$, где Λ — максимальное (по модулю) собственное значение матрицы $f_x(t)$ (для системы $\dot{x} = f(x)$), а величина Λ — того же порядка, что и коэффициенты системы (1). В принятой

в расчетах схеме обеспечить необходимую точность интегрирования было несложно, хотя шаг сетки для u был с этой точки зрения очень велик: $\Delta t = T/50 \gg dt$. Однако, кроме этого, есть и еще одно неприятное обстоятельство: решение сопряженной к (1) системы

$$\begin{aligned} -\dot{\psi}_1 &= -(C + Du)\psi_1, \\ -\dot{\psi}_2 &= A\psi_1 - A\psi_2 \end{aligned} \quad (17)$$

— суть быстро убывающие (влево) функции типа экспонент. Поэтому и функциональные производные

$$\frac{\partial x^1(\tau_i)}{\partial u(t)} = w_i(t) \quad (18)$$

имеют характер, качественно показанный на рис. 45. На значение $x^1(\tau_i)$ влияет в основном лишь вариация $\delta u(t)$ в малой окрестности точки аппроксимации τ_i ; вариации же управления вне малых окрестностей точек τ_i почти не повлияют на значения $x(\tau_i)$, входящие в аппроксимацию (16), но существенно повлияют на значение δF_0 . Замена $\delta \dot{u} = \delta v$ меняет ситуацию: функциональная производная $\tilde{w}_i(t) = \partial x^1(\tau_i) / \partial v(t)$ показана на том же рис. 45. Можно еще иначе трактовать эту замену: в процессе варьирования $u(t) \rightarrow u(t) + \delta u(t)$ используются функции $\delta u(t)$, малые

не только в норме $\max_t |\delta u(t)|$, но и в норме $\max_t |\delta \dot{u}(t)|$. Во всяком случае, попытки решения задачи в постановке (1)–(4) при 11 точках аппроксимации в задачах с большими z оказались неудачными, хотя трудоемкость задачи увеличилась — увеличился и вертикальный размер задачи линейного программирования, увеличилась и трудоемкость вычисления функциональных производных, так как теперь сопряженную систему нужно было интегрировать 11 раз, а не три, как было раньше. Разумеется, этот прием имеет и отрицательные последствия: вместо разрывов в $u(t)$ нужно получить δ -функции в $v(t)$. Соответствующие дефекты численного решения («размазанные» разрывы в u) хорошо видны на рис. 44.

В а р и а н т ы з а д а ч и о б о с т а н о в к е р е а к т о р а. Задача решалась и в других постановках. Например, время T считалось не фиксированным и входило в обобщенное управление — комплекс $\{u(\cdot), T\}$, который нужно было определить решением задачи: найти

$$\min_{u(\cdot), T} F_0[u(\cdot), T], \quad \text{где } F_0[u(\cdot), T] \equiv T, \quad (19)$$

при условиях

$$F_1[u(\cdot), T] \leq X_1, \quad \text{где } F_1 \equiv \max_{0 \leq t \leq T + \Delta T} x^1(t), \quad (20)$$

$$0 \leq u(t) \leq 1.$$

Здесь X_1 — заданное число, ΔT (время проведения работ) также задано. Таким образом, мы имеем задачу быстрогодействия с фазовым ограничением. Решение ее осуществлялось примерно так же, как это описано выше (см. [3]), и само оптимальное управление имеет такой же характер (13). Решалась задача и для очень мощных аппаратов, в которых, кроме отравления ксеноном, нужно было учитывать и отравление другим элементом (самарием). Система уравнений (1) расширялась добавлением еще двух:

$$\begin{aligned} \dot{x}^3 &= c(u - x^3), & x^3(0) &= 1, \\ \dot{x}^4 &= a(x^3 - ux^4), & x^4(0) &= 1. \end{aligned} \quad (21)$$

Менялось и выражение для F_0 :

$$F_0[u(\cdot)] \equiv \max_{0 \leq t \leq \infty} [x^1(t) + lx^4(t)] \quad (22)$$

(a, c, l — заданные числа). Заметим, что бесконечность интервала времени здесь существенна, так как, в отличие от $x^1(t)$, $\lim_{t \rightarrow \infty} x^4(t) > 0$. Оптимальное управление в этой задаче оказалось сложнее, структуру точного решения угадать не удалось. На рис. 46 показана функция $\varphi(t) \equiv x^1(t) + lx^4(t)$ для численно оптимального $u(t)$. Характерно, что множество $M_0 = \{t : \varphi(t) =$

$= \max_t \varphi(t)$ состоит из четырех изолированных частей: из отрезка $[t'', t''']$ и трех изолированных точек $t' < t''$, $t^* > T$ и $t = \infty$ *). С бесконечностью и здесь никаких проблем нет, так как $\varphi(\infty)$ легко вычисляется аналитически через $x(T)$. Эта задача решалась так же, как и первоначальная, но на машине в три раза более быстрой. Число точек аппроксимации для F_0 было увеличено до 5—6. Что касается ограничений $0 \leq u(t) \leq 1$, превращающихся после замены $\dot{u} = v$ в фазовые ограничения (8), то на границы 0 и 1 $u(t)$ в процессе поиска не вышло. Дело в том, что для

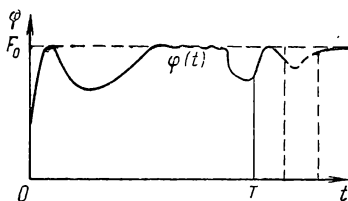


Рис. 46.

этих задач было характерно необычайно сильное влияние вариации $\delta u(t)$ на δF_0 . Поэтому интервал, на котором в точном решении $u(t) = 1$, был настолько мал, что его не удалось покрыть несколькими счетными интервалами сетки для u (в этих расчетах она содержала 100 точек на $[0, T]$), хотя сетка была взята неравномерной: шаг был существенно мень-

шим $T/100$ в начале и в конце интервала $[0, T]$. На рис. 46 в действительности изображена функция $\varphi(n)$, где n — номер точки сетки для управления.

История задачи. Вопрос о постепенном выключении реактора был сначала поставлен физиками (см. [22], стр. 377) и решался типично инженерными средствами: в [98] строилось трехпараметрическое семейство управлений $u(t; \alpha_1, \alpha_2, \alpha_3)$ и решалась (численно) задача определения $\min_{\alpha} \max_t x^1(t)$.

В принципе, при подходящей конструкции семейства $u(t; \alpha)$ таким образом можно получить почти точное решение. Но в [98], при отсутствии информации о решении, семейство $u(t; \alpha)$ содержало лишь монотонно убывающие функции, которыми никак нельзя аппроксимировать решение (13); поэтому эффект такого «оптимального» управления был незначителен (по сравнению с тривиальным выключением). Четкая постановка задачи о выключении как вариационной была дана Р. Беллманом [71]; был предложен и алгоритм ее решения, использующий идеи динамического программирования (см. также [8], [9], [41]). В дальнейшем в монографии [4], специально посвященной этой задаче, были опубликованы данные о реализации этой программы. Они заслуживают подробного комментария. Заметим еще, что вычислительная схема

*) Здесь имеется в виду, что $\lim_{t \rightarrow \infty} \varphi(x) = \max_t \varphi(x)$.

метода [4] имеет много общего со схемой § 15, однако в некоторых пунктах существенно от нее отличается.

Решение методом динамического программирования. В [4] алгоритм динамического программирования был использован для решения следующей конкретной задачи. Система уравнений имела вид

$$\begin{aligned} \dot{x}^1 &= f^1(x, u): \quad x^1 = -0,724x^1 - 20ux^1 + 19,67x^2 + 1,05u; \\ \dot{x}^2 &= f^2(x, u): \quad x^2 = -x^2 + u; \\ x^1(0) &= 1; \quad x^2(0) = 1; \end{aligned} \quad (23)$$

$$0 \leq u(t) \leq 2 \quad \text{при} \quad 0 < t < T = 1; \quad u(t) \equiv 0 \quad \text{при} \quad t > T.$$

Непосредственно решать задачу на $\min_u \max_t x^1(t)$ алгоритм не позволяет, поэтому рассматривается другая: найти

$$\min x^1(T_0) \quad \text{при условии} \quad \max_{t < T} x^1(t) \leq x_c, \quad (24)$$

где $T_0 > T$ либо задано, либо $T_0 = \arg \max_{t > T} x^1(t)$.

Решая (24) с разными x_c и добиваясь (подбором) совпадения: $x_c = \max_{t > T} x^1(t)$, можно решить и исходную задачу. На интервале управления $[0, T]$ вводится равномерная сетка точек $t_0, t_1, \dots, t_N = T$ (в [4] $N=20$), в каждой точке определен экземпляр сетки в фазовом пространстве $\{x^1, x^2\}$, покрывающий область возможных значений x^1, x^2 (здесь она легко оценивается: $0 \leq x^1 \leq x_c, 0 \leq x^2 \leq 2$). В [4] сетки имели по 30×30 узлов. На каждой сетке S_n определена функция $F_n(x^1, x^2)$.

Эта функция (функция Беллмана) имеет простой содержательный смысл. Пусть в момент t_n система находится в состоянии $\{x^1, x^2\}$, и определяется оптимальное управление на отрезке (t_n, T) . Тогда $F_n(x^1, x^2)$ есть минимальное значение $x^1(T_0)$. Функции F_n последовательно вычисляются для $n=N, N-1, \dots, 1$, с помощью уравнения динамического программирования (см. 44).

1. Начальные данные: на интервале $[T, T_0]$ при $u \equiv 0$ система (23) элементарно интегрируется и находится функция

$$x^1(T_0) = \Phi[x^1(T), x^2(T)].$$

Тогда $F_N(x^1, x^2) = \Phi[x^1, x^2]$.

2. Определяется в узлах сетки функция $F_{N-1}(x^1, x^2)$ из уравнения

$$F_{N-1}(x^1, x^2) = \min_{0 \leq u \leq 2} F_N[x^1 + dt \cdot f^1(x, u), x^2 + dt \cdot f^2(x, u)]. \quad (25)$$

где $f^i(x, u) = f^i(x^1, x^2, u)$ — правые части (23). Анализ системы с помощью принципа максимума показывает, что оптимальное управление релейно ($u=0$ или 2). Поэтому в (25) нужно сравнить

только два значения. Разумеется, точки $x^1 + dt \cdot f^1$, $x^2 + dt \cdot f^2$ не попадают в узлы сетки S_N , поэтому необходимые значения интерполируются.

3. Таким же образом находятся сеточные функции

$$F_{N-2}, F_{N-3}, \dots, F_1(x^1, x^2).$$

4. Определяется оптимальное управление $u_{n+1/2}$ на интервалах (t_n, t_{n+1}) . Сначала при заданных x_0^1, x_0^2 , решаем задачу: найти

$$\min_{0 \leq u \leq 2} F_1[x_0^1 + dt f^1(x_0, u), x_0^2 + dt \cdot f^2(x_0, u)], \quad (26)$$

и получаем $u_{1/2}$ ($=0$ или 2). Однако в (25) используется слишком грубая схема численного интегрирования, поэтому для определения $x_1^1 = x^1(t_1)$, x_1^2 система (23) интегрируется с найденным $u_{1/2}$ с необходимой точностью.

5. После определения x_1^1, x_1^2 из уравнения типа (26) находится оптимальное $u_{1/2}$; далее интегрированием (23) находятся x_2^1, x_2^2 и т. д.

Таким образом, получено управление ([4], стр. 95):

$$u(t) = \begin{cases} 0 & \text{при } t \in (0; 0,7) \cup (0,75; 0,9) \\ 2 & \text{при } t \in (0,70; 0,75) \cup (0,9; 1,0). \end{cases} \quad (27)$$

К сожалению, по приведенным в [4] данным трудно понять, какая именно задача решалась. По некоторым признакам T_0 было задано, однако для него приведено значение $T_0 = T = 1$, что совершенно бессмысленно. Мы будем анализировать результаты так, как если бы решалась задача на $\min_{u, t \geq T} x^1(t)$. Если в действительности решалась задача с заданным T_0 в диапазоне, например, $1,3T \leq T \leq 3T$, выводы были бы такими же (даже количественно). Оптимальные управления для всех таких задач почти не отличаются друг от друга.

Прежде всего рассмотрим возможные источники ошибки приближенного решения (а здесь, как мы увидим, ошибка по значению минимизируемого функционала $\approx 20\%$):

1) ошибочное предположение о релейном характере управления;

2) грубость сетки с шагом $dt = T/20$;

3) использование в уравнении (25) грубой разностной схемы;

4) интерполяция $F_n(x)$ в уравнениях (25), (26).

Траектория, соответствующая управлению (27), такова, что при $t=0,7$ $x^1(t)=8,145$, т. е. достигает значения $x_s=8,15$. Поэтому первый импульс в u необходим, иначе будет нарушено ограничение $x^1 \leq 8,15$. Однако предположение о релейном характере управления при столь большом шаге сетки ($dt=0,05$)

привело к существенному завышению величины импульса:

$$\int_{0,75}^{0,7} u dt = 0,1. \text{ Заметим, что само по себе предположение о релейности,}$$

даже если оно ошибочно, не исключает возможности получения достаточно точного решения, если управление входит в задачу линейно: любое $u(t)$ может быть сколь угодно точно аппроксимировано релейным. Но точность такой аппроксимации существенно зависит от шага сетки по t . Более того, даже если оптимальное управление действительно релейно (а в данной задаче есть и участок релейного управления — второй импульс), с вычислительной точки зрения лучше не связывать себя этим ограничением. Ведь при сеточном описании измеримых функций $u(t)$ есть два способа трактовать таблицу значений u :

$$u_n = u(t_n) \text{ и } u_{n+1/2} = \frac{1}{t_{n+1} - t_n} \int_{t_n}^{t_{n+1}} u dt,$$

и в вопросах управления системами дифференциальных уравнений второй способ более естествен: на траекторию влияют не мгновенные значения u , а средние по малым интервалам. Мы имели уже случай продемонстрировать это в § 27, и здесь тоже будет показано, что, отказавшись от релейности, мы расширяем возможности аппроксимации.

Отметим основное отличие данной реализации метода динамического программирования от схемы вычислений § 15. Оно связано с использованием интерполяции функции Беллмана $F(x^1, x^2)$ с узлов сетки. Этим снимается ограничение на шаг сетки в фазовом пространстве типа $h=0(\tau)$, необходимое в схеме метода Н. Н. Моисеева. Вместе с тем интерполяция является источником определенных ошибок, тем более, что сетки приходится брать сравнительно грубые. Кроме того, используя интерполяцию, неявно предполагают наличие у функции Беллмана таких свойств гладкости, которых может и не быть. Известны простые примеры задач, в которых функция Беллмана разрывна, а наличие разрывов производной может считаться почти общим явлением. Схема вычислений § 15 может быть (при $h=O(\tau^2)$) обоснована без всяких предположений о свойствах функции Беллмана. Что касается реализации алгоритма на ЭВМ, то в данном случае наибольшие ограничения связаны с ресурсом памяти. Вычисления в [4] требуют N таблиц по 30×30 величин, однако при вычислении очередной функции $F_n(x^1, x^2)$ в оперативной памяти нужно иметь только две такие таблицы.

Для управления (27) $\max_{t \geq T} x^1(t) = 5,522$. Задача была решена методом последовательной линеаризации (§§ 19—21). В качестве

начального приближения бралась функция $u(t) = 0,1$, процесс решения показан в табл. 1 для сетки с $N = 100$ (время по БЭСМ-6

Т а б л и ц а 1

$N = 20$			$N = 100$		
ν	$\max_{t < T} x^1(t)$	$\max_{t > T} x^1(t)$	ν	$\max_{t < T} x^1(t)$	$\max_{t > T} x^1(t)$
0	4,45	5,60	0	4,45	5,60
5	5,15	4,89	5	5,296	4,886
10	7,04	4,72	10	7,080	4,788
15	7,25	4,67	15	7,207	4,672
20	7,75	4,63	20	7,675	4,633
25	8,03	4,62	25	7,865	4,614
30	8,06	4,60	30	8,169	4,600
35	8,15	4,594	35	8,060	4,593
40	8,15	4,585	40	8,146	4,586

около 2,5 минут) и для сетки с $N = 20$ (0,5 минут). В таблице 2 представлены: ν — номер итерации, $\max_{t < T} x^1(t)$ и $\max_{t > T} x^1(t)$. Анализ численных результатов позволяет предложить простые аппроксимации оптимального управления:

$$\text{I. } u(t) = \begin{cases} 0 & \text{при } t \in (0; 0,7) \cup (0,7032; 0,95), \\ 2 & \text{при } t \in (0,7; 0,7032) \cup (0,95; 1,0); \end{cases}$$

$$\text{II. } u(t) = \begin{cases} 0 & \text{при } t \in (0; 0,7) \cup (0,75; 0,95), \\ 0,128 & \text{при } t \in (0,7; 0,75), \\ 2 & \text{при } t \in (0,95; 1,0); \end{cases}$$

$$\text{III. } u(t) = \begin{cases} 0 & \text{при } t \in (0; 0,7), \\ 0,025 & \text{при } t \in (0,7; 0,95), \\ 2 & \text{при } t \in (0,95; 1,0). \end{cases}$$

Все они объединены общими характеристиками: величина второго импульса: $\int_{0,95}^{1,0} u dt = 0,1$, и величина первого импульса равна 0,0064, причем в III этот импульс растянут до $t = 0,95$. Эти решения представлены в табл. 2, I — III (последнее в таблице значение $t = \arg \max_{t > T} x^1(t)$).

Решение методом математического программирования. Для той же системы (23) решение вариационной задачи было опубликовано в [75] (1965 г.); затем оно было подробно освещено в [76] (русский перевод [77]). Эта

Т а б л и ц а 2

I						
t	0	0,7	0,7032	0,95	1,0	2,04
x^1	1,0	8,157	7,197	7,990	1,279	4,546
x^2	1,0	0,4966	0,5014	0,3917	0,4702	0,1660
u	0	2	0	2	0	
II						
t	0	0,7	0,750	0,95	1,0	2,02
x^1	1,0	8,157	7,371	7,964	1,275	4,545
x^2	1,0	0,4966	0,4786	0,3919	0,4703	0,1695
u	0	0,128	0	2	0	
III						
t	0	0,7	0,950	1,0	2,04	
x^1	1,0	8,157	7,878	1,264	4,543	
x^2	1,0	0,4966	0,3923	0,4707	0,1663	
u	0	0,025	2	0		

работа заслуживает подробного комментария как характерный пример, подтверждающий тривиальность решения задач оптимального управления «в принципе» и сложность их фактического решения. В [77] предлагается и используется следующая простая схема решения. На интервале $[0, T]$ вводится сетка с переменным шагом:

$$t_0 = 0, \quad t_1 = t_0 + T_{1/2}, \dots, t_{n+1} = t_n + T_{n+1/2}, \dots$$

Дифференциальные уравнения заменяются разностными:

$$x_{n+1} = x_n + T_{n+1/2} \cdot f(x_n, u_{n+1/2}), \quad n = 0, 1, \dots, N-1. \quad (28)$$

В [77] решается задача быстродействия при ограничении в фазовом пространстве $\max_t x^1(t) \leq x_c$. Это условие, так же как и ограничение $0 \leq u \leq 2$, очевидным образом порождает ограничения

$$x_n^1 \leq x_c; \quad 0 \leq u_{n+1/2} \leq 2. \quad (29)$$

Таким образом, получаем задачу математического программирования: найти

$$\min \sum_{n=0}^{N-1} T_{n+1/2} \text{ при условиях (28), (29).} \quad (30)$$

Искомymi переменными являются: x_n^1 , x_n^2 , $u_{n+1/2}$, $T_{n+1/2}$. В [77] $N=12$, и решение вариационной задачи свелось к дискретной задаче с 48 переменными, с 24 условиями-равенствами (28) и 36-ю условиями-неравенствами (29). Это — изученная задача, для ее решения разработано большое число алгоритмов, включенных в систему математического обеспечения современных ЭВМ. Остается воспользоваться такой программой. Именно так и решается задача в [77], причем используется программа *безусловной минимизации*: с помощью штрафных функций задача сводится к минимизации одной функции от 48 переменных. Минимум ищется каким-то вариантом спуска по градиенту (в других местах [77] упоминается обобщенный метод Ньютона, в котором используется матрица вторых производных минимизируемой функции). За 8 минут работы IBM-7094 было получено решение, представленное заимствованной из [77] (стр. 150) табл. 3.

Т а б л и ц а 3

n	$T_{n-1/2}$	$u_{n-1/2} \cdot 10^3$	x_n^1	x_n^2	n	$T_{n-1/2}$	$u_{n-1/2} \cdot 10^3$	x_n^1	x_n^2
1	0,032	4,67	1,001	1,000	7	0,032	4,67	1,007	1,000
2	0,032	4,67	1,002	1,000	8	115,035	0,07	4,796	0,799
3	0,032	4,67	1,003	1,000	9	0,034	2,14	4,737	0,799
4	0,032	4,67	1,004	1,000	10	0,032	2,14	4,797	0,799
5	0,032	4,67	1,005	1,000	11	0,032	2,14	4,798	0,799
6	0,032	4,67	1,006	1,000	12	533,921	129,05	4,800	0,177

Обратите внимание на вырожденность временной сетки: по существу, $[0, T]$ разбит только на два счетных интервала, остальные — практически нулевые. Этот же дефект имеет решение второго варианта задачи ([77], таблица на стр. 151). Никакого отношения к решению дифференциальных уравнений табличные функции не имеют. Любопытно, что одно из таких «решений» было проконтролировано расчетом с $N=20$, получено совпадение по функционалу с точностью до 0,15%, и можно утверждать, что «ошибка дискретизации является допустимой для практики» ([77], стр. 151). Это совпадение связано, видимо, с тем, что и сетка с $N=20$ столь же вырождена и состоит из тех же двух счетных интервалов.

Содержательное обсуждение подобных решений бессмысленно, здесь нельзя даже говорить о какой-то, пусть не очень высокой, точности. Характерно, что используемый метод полностью обосно-

ван: есть теоремы о том, что решение разностной задачи (28) при достаточно большом N аппроксимирует решение исходной задачи, есть теорема о том, что минимум составного функционала метода штрафных функций (при достаточно больших коэффициентах штрафа) аппроксимирует решение дискретной задачи (28). Есть, наконец, и теорема о сходимости (при достаточно большом числе итераций) используемого в стандартной программе метода поиска минимума. Но если это так, значит, можно как-то исправить положение и получить этим методом приемлемые результаты. Что же для этого нужно сделать? Прежде всего, ввести ограничение на шаги типа $T^- \leq T_{n+1/2} \leq T^+$, предупреждающие вырождение сетки. Далее, число N следует увеличить. Легко оценить требуемый для точности (допустим, 1%) шаг интегрирования. Для схемы первого порядка точности относительная ошибка численного интегрирования имеет величину $|\Delta dt|$, где Λ — максимальное по модулю собственное число матрицы f_x ; оно здесь легко оценивается: при $u=0$, $\Lambda=1$; при $u=2$, $\Lambda=40$. Таким образом, число N нужно увеличить хотя бы в 10 раз, а если рассчитывать (не зная заранее, какие u появятся в процессе решения) и на ситуации $u=2$, следует взять $N \approx 1000$ *). Если считать, что при увеличении N в 10 раз объем вычислений возрастет в 100 раз (а это еще оптимистический прогноз), решение потребует ≈ 1000 минут работы машины класса БЭСМ-6. И еще не ясно, что из этого получится. Выше уже упоминалось, что на сетке с $N=100$ (а численное интегрирование велось с еще меньшим шагом) автор решал подобные задачи на БЭСМ-6 за 2,5 минуты, а при сетке с $N=20$ — за 0,5 минуты с точностью (по функционалу), вероятно, $\sim 1\%$.

Аналитические исследования задачи с помощью принципа максимума проводилось неоднократно и были достаточно успешными. Первая, видимо, из таких попыток была предпринята в 1964 г. (результаты наших расчетов, содержащие и гипотезу о точном решении, были опубликованы в том же году в [2], [87]). Любопытно, что было получено другое решение, имеющее структуру (1721);

$$u(t) = \begin{cases} 0 & \text{при } 0 < t < t_1, \\ 2 & \text{при } t_1 < t < t_2, \\ 0 & \text{при } t_2 < t < t_3, \\ 2 & \text{при } t_3 < t < T. \end{cases} \quad (31)$$

Соответствующая этому управлению функция $x'(t)$ имеет три точки

*) Контролируя решение (27), автор интегрировал систему (23) с разными шагами. При шаге $dt=1/200$ ошибка в некоторых точках достигала 1%, хотя использовалась схема второго порядка точности.

локальных максимумов, t_1 , t_3 и $t^* > T$, причем $\max_t x'(t) = x^1(t_2) = x^1(t^*) > x^1(t_1)$. Этот режим является обобщением режима (12) и существует в узком диапазоне времени управления

$$T_{кр}(z) < T \leq T'_{кр}(z), \quad T'_{кр}(z) - T_{кр}(z) \ll T_{кр}(z).$$

Автором была проведена проверка, и оказалось, что решение (31) действительно удовлетворяет принципу максимума, но дает значение $\max_t x^1(t)$ несколько большее, чем решение (13); однако разница крайне незначительная, и основным недостатком (24) является очень узкий диапазон существования. При $T \rightarrow T'_{кр}(z)$ значения $x^1(t_2) = x^1(t^*)$ падают, $x^1(t_1)$ растет, при $T = T'_{кр}(z)$ они сравниваются. Видимо, при $T > T'_{кр}(z)$ можно искать оптимальные режимы с тремя импульсами и т. д.

В дальнейшем в работах [71], [20] (1965 г.) с помощью принципа максимума было получено и решение (13). Любопытный и, в сущности, единственный известный автору пример прикладной задачи, в которой найдено два локальных минимума. Кстати для управления [31] принцип максимума является не только необходимым, но и достаточным условием минимума (локального, разумеется). Этот факт аналогичен тому, что для функции $y(u) \equiv u$ на интервале $0 \leq u \leq 1$ необходимое условие минимума является в то же время и достаточным.

§ 37. Задача о спуске космического аппарата

Вход управляемого космического аппарата в плотные слои атмосферы и спуск его на поверхность Земли в некотором приближении описываются следующей системой дифференциальных уравнений:

$$\begin{aligned} \frac{dx^1}{dt} &= x^2, \\ \frac{dx^2}{dt} &= x^1(x^4)^2 - \frac{\mu}{(x^1)^2} - Q(x^1, x^2, x^4)(x^2 - ux^1x^4), \\ \frac{dx^3}{dt} &= x^4, \\ \frac{dx^4}{dt} &= \frac{2x^2x^4}{x^1} - Q(x^1, x^2, x^4)\left(x^4 - u\frac{x^2}{x^1}\right), \\ 0 &\leq t \leq T, \end{aligned} \tag{1}$$

где $Q(x^1, x^2, x^4) = Ce^{\frac{h-x^1}{H}} \sqrt{(x^2)^2 + (x^1x^4)^2}$. Функция $u(t)$ и время T — искомые элементы управления. Физический смысл входящих в задачу величин см. в [19].

В момент $t=0$ заданы начальные значения $x^1(0)$, $x^2(0)$, $x^3(0)$, $x^4(0)$, образующие группу условий $\Gamma x=0$. Определены

функционалы

$$F_0[u(\cdot), T] \equiv \max \Phi[x(t)] \equiv \max_t Ce^{\frac{h-x^1(t)}{H}} \{[x^2(t)]^2 + [x^1(t)x^4(t)]^2\} \quad (2)$$

($F_0[u(\cdot), T]$ имеет смысл максимальной перегрузки),

$$F_1[u(\cdot), T] \equiv x^1(T) - R_3, \quad (3)$$

$$F_2[u(\cdot), T] \equiv x^3(T) - \Psi. \quad (4)$$

Задача состоит в определении управления $\{u(\cdot), T\}$, стесненного условием

$$u(t) \in U: \quad |u(t)| \leq 0.2,$$

и минимизирующего значение F_0 при условиях $F_1=0$, $F_2=0$, определяющих попадание аппарата в заданную точку поверхности Земли. Числовые значения входящих в задачу постоянных взяты из работы [19]:

$$\mu = 398\,600; \quad c = 2; \quad H = 7; \quad h = 6384; \quad R_3 = 6371; \quad x^1(0) = 6471.$$

Что касается остальных начальных данных, то рассматриваются два варианта задачи:

а)

$$x^2(0) = -0,008725; \quad x^3(0) = -0,157368; \quad x^4(0) = 0,00171 \quad (\rho_\pi = 60);$$

б)

$$x^2(0) = -0,007559; \quad x^3(0) = -0,13632; \quad x^4(0) = -0,001711 \quad (\rho_\pi = 70).$$

Величина Ψ в различных расчетах принимает разные значения, в своем месте они будут указаны. Задача методически интересна тем, что в ней присутствуют многие характерные особенности, осложняющие численное решение: в правой части (2) максимум достигается на некотором интервале $[t_2, t_3]$, причем $(t_3 - t_2) \sim \frac{1}{2} T$; на этом же интервале в управлении реализуется так называемый «особый режим» ($|u(t)| < 0,2$) на остальной части $[0, T]$ управление «релейное» ($|u(t)| = 0,2$). В этой задаче известно точное решение. Оно найдено в [19] в результате анализа полученных численных решений. Точнее говоря, по численным результатам угадана структура решения, после чего уже более точными расчетами находится траектория, имеющая данную структуру. Все это ниже подробно объясняется, а угадать структуру решения читатель сможет и сам, познакомившись с изображенными на рисунках 50, 51 приближенными оптимальными управлениями. Впервые численное решение задачи было осуществлено в [19] комбинацией следующих вычислительных приемов,

I. Задача на $\min_u \max_t \Phi[x(t)]$ сводится к последовательности терминальных задач с фазовым ограничением. Вводится параметр α , и ищется управление $u(\cdot)$ из условий:

$$F_0[u(\cdot), T] \leq \alpha; \quad F_1[u(\cdot), T] = 0; \quad F_2[u(\cdot), T] = 0. \quad (5)$$

При одних значениях α задача (5) имеет решение, при других — нет, и исходная задача сводится к определению минимального α , при котором (5) имеет решение. Минимальное α ищется процессом типа «деления вилки», каждый акт которого требует решения терминальной задачи (5). Эта редукция указана в работе автора [92], однако никогда им не использовалась и не рекомендовалась, так как методом последовательной линеаризации можно прямо решать исходную задачу на $\min F_0[u(\cdot), T]$.

II. Задача (5) методом штрафных функций заменяется задачей на $\min F_\alpha[u(\cdot), T]$, где *

$$F_\alpha[u(\cdot), T] \equiv \int_0^T \{\Phi[x(t)] - \alpha\}_+^2 dt + \mu_1 F_1^2[u(\cdot), T] + \mu_2 F_2^2[u(\cdot), T]. \quad (6)$$

В [92] были указаны и возможные затруднения в реализации этого подхода: задача (5) решается итерациями, и по эволюции F_α в процессе итераций нужно делать вывод о том, какая ситуация имеет место: $\min F_\alpha[u(\cdot), T] = 0$ или $\min F_\alpha[u(\cdot), T] > 0$. Когда α далеко от искомого минимального, такой вывод делается легко и достаточно надежно, но при α близких к минимальному задача идентификации затрудняется. К сожалению, в [19] нет сведений о том, каких затрат требует определение α с нужной точностью. Подробно приведены лишь данные о решении задачи (5) с уже определенным α , причем исходное управление выбирается без учета уже найденных при выборе α траекторий. Этот расчет можно трактовать просто как самостоятельное решение задачи (5).

III. Решение задачи (6) осуществляется некоторым вариантом градиентного спуска. Он заслуживает пояснения. Один шаг спуска состоял из следующих вычислений.

1. С известным $u(t)$ интегрировались уравнения (1) (условие $F_1[u(\cdot), T] = 0$ или $F_2[u(\cdot), T] = 0$ использовалось как признак окончания интегрирования; им определялось T).

2. Находилась функция $\phi(t)$ интегрированием сопряженного уравнения. Правая часть и начальные данные (при $t=T$) этого уравнения выбираются так, что $\phi(t)$ позволяет вычислить производную функционала $F_\alpha[u(\cdot), T]$. Образуется функция Гамильтона $H(x, u, \phi)$. Так как система (1) линейна по u , то

*) Под $\{a\}_+$ понимается $\max\{a, 0\}$.

$H(x, u, \psi) = H_0(x, \psi) + uH_1(x, \psi)$ (заметим, что $\partial H / \partial u = H_1(x, \psi) = \partial F / \partial u(\cdot)$). Далее функция $u(t)$ варьировалась, причем использовался класс конечных вариаций на множестве малой меры. Выбиралось число Δ , и находилось (подбором) $\varepsilon(\Delta) > 0$ так, что мера множества $\{t: H[x(t), u(t), \psi(t)] \geq \max_t H - \varepsilon(\Delta)\}$ равна Δ .

На этом множестве $u(t)$ заменялось на $u^*(t)$, определяемое условием $H[x(t), u^*, \psi(t)] = \max_t H[x(t), u, \psi(t)]$ (в данном случае $u^* = 0,2$ ($-0,2$) при $H_1 > 0$ (< 0)).

Т а б л и ц а 1

ν	F_α	F^*	ΔL	μ
0	7881,1	6884,2	-315,8	0,1
5	455,8	407,6	-69,4	
9	39,8	32,9	25,6	
10	35,99	33,05	17,09	
25	23,74	18,95	21,87	
150	4,63	2,536	14,47	0,01÷1,00
270	1,48	0,409	10,36	
360	0,171	0,107	2,52	

3. Новое управление определяется числом Δ ; этим же числом определяется и новое значение функционала F_α . Теперь возникает задача определения (или назначения) Δ . В [19] сказано, что Δ выбиралось из условия минимизации F_α . Видимо, это следует понимать так: для нескольких Δ находилось значение $F_\alpha(\Delta)$. Для этого при каждом Δ нужно найти $\varepsilon(\Delta)$, $u(\Delta)$ и, проинтегрировав систему (1), найти $F_\alpha(\Delta)$. Далее, тем или иным способом находится (не очень точно) $\min_\Delta F_\alpha(\Delta)$. В [19] нет данных о трудоемкости этого подбора. В табл. 1 (заимствованной из [19]) процесс решения показан следующими величинами: ν — номер итерации, F_α ,

$$F^* \equiv \int_0^T \{\Phi[x(t)] - \alpha\}_+^2 dt \quad (7)$$

— характеристика суммарного нарушения условия $\Phi[x(t)] \leq \alpha$; $\Delta L = R_3 F_2$ (задача решалась при $L=500$, т. е. $\Psi=500/R_3$, начальные данные б)). В табл. 1 приведена и величина штрафного коэффициента μ_1 ; на последнем этапе расчета его пришлось варьировать, чтобы получить хорошую точность. Подбор μ_1 , видимо, производился вручную, так как никаких указаний на алгоритм подбора нет. Заметим, что используемый в [19] метод

варьирования управления приводит к тому, что оно на каждом этапе имеет релейный характер ($|u(t)| = 0,2$). Точное решение имеет и участок, на котором $|u(t)| < 0,2$. Очевидно, однако, что в силу линейной зависимости правых частей (1) от u такое решение может быть сколь угодно точно (по величинам входящих в задачу функционалов) аппроксимировано релейным.

Точное решение задачи. На основании численных расчетов была угадана структура точного решения задачи, после чего найти его уже нетрудно, — тоже, конечно, с использованием численных методов, но гораздо более простых и точных, чем итерационные методы решения вариационных задач. Структура эта такова: интервал $[0, T]$ разбивается на четыре части точками $0 < t_1 < t_2 < t_3 < T$; оптимальное управление имеет форму

$$u(t) = \begin{cases} -0,2 & \text{при } t \in (0, t_1), \\ 0,2 & \text{при } t \in (t_1, t_2), \\ u^*(t) \in (-0,2; 0,2) & \text{при } t \in (t_2, t_3), \\ -0,2 & \text{при } t \in (t_3, T). \end{cases} \quad (8)$$

При этом (см. рис. 49) определяющая функционал F_0 функция $\Phi[x(t)]$ ведет себя на оптимальной траектории следующим образом:

$$\Phi[x(t)] \begin{cases} < \max_t \Phi[x(t)] & \text{при } t \in (0, t_2) \cup (t_3, T), \\ = \max_t \Phi[x(t)] & \text{при } t \in [t_2, t_3]. \end{cases} \quad (9)$$

Моменты t_1, t_2, t_3, T и функцию $u^*(t)$, однозначно определяемые значением параметра α , удобно находить, взяв в качестве независимого аргумента t_1 , а не α . Вычисления организуются по следующей схеме.

1. Назначается некоторое число t_1 .

2. Система уравнений (1) интегрируется (численно) с заданными начальными значениями $x(0)$ и с управлением $u(t) = -0,2$ на отрезке $[0, t_1]$.

3. Далее, при $t > t_1$ интегрирование продолжается с $u(t) \equiv +0,2$; момент t_2 определяется следующими соображениями.

Вводится функция

$$\dot{\Phi}[x(t)] \equiv \Phi_x[x(t)] \frac{dx}{dt}, \quad (10)$$

причем $\frac{dx}{dt}$ заменены правыми частями (1). $\dot{\Phi}(x)$ имеет довольно громоздкое выражение, но для дальнейшего важно лишь то, что в правую часть (10) u явно не входит. Определим функцию

$$\ddot{\Phi}[x(t), u(t)] \equiv \dot{\Phi}_x[x(t)] \frac{dx}{dt}. \quad (11)$$

В эту функцию u уже входит явно, причем линейно, в силу линейности правых частей (1) по u . Таким образом, Φ имеет вид $\Phi(x, u) = \Phi_0(x) + u\Phi_1(x)$. Аналитические выражения функций Φ_0 , $\Phi_1(x)$ могут быть получены тривиальными, но утомительными формальными выкладками. На отрезке $[0, t_1]$ имеем $\dot{\Phi}[x(t)] > 0$; это соотношение сохраняется и при $t > t_1$, однако значение $\Phi[x(t)]$ падает, и в качестве t_2 берется момент времени, в который $\dot{\Phi}$ обращается в 0. В этот момент t_2 функция $\Phi[x(t)]$ достигает своего максимального на данной оптимальной траектории

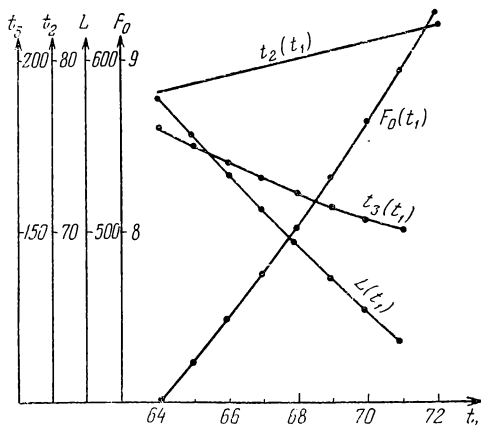


Рис. 47.

значения F_0 ; на следующем интервале $[t_2, t_3]$ функция $\Phi[x(t)] = F_0 (= \alpha)$. На $[t_2, t_3]$ постоянство $\Phi[x(t)]$ обеспечивается выбором управления из уравнения $\dot{\Phi}[x(t), u(t)] = 0$, т. е.

$$u^*(t) = -\Phi_0[x(t)]/\Phi_1[x(t)]. \quad (12)$$

4. Момент t_3 определяется условием $u^*(t_3) = -0,2$.

5. При $t_3 \leq t \leq T$ полагаем $u(t) = -0,2$, причем T определяется из условия $x^1(T) = R_3$. Таким образом, параметром t_1 однозначно определяются значения $t_2, t_3, T, F_0 (= \alpha)$ и оптимальная управляющая функция $u^*(t)$. На рис. 47 показаны графики величин t_2, t_3, L, F_0 в зависимости от t_1^*). Они построены интерполяцией по значениям для дискретного набора t_1 . Этот график соответствует задаче с начальными данными а). Что касается оптимальных функций $u^*(t)$, то они будут сравниваться с теми, которые получаются в результате приближенного решения задачи методами спуска в пространстве управлений (см. рисунки 50, 51).

*) Здесь $L = R_3 x^3(T)$.

Заметим, что при реализации описанного выше построения точного решения нужно особое внимание обратить на определение момента t_2 : при численном интегрировании системы (1) с шагом dt обычно реализуется ситуация

$$\Phi[x(t^k)] > 0, \quad \Phi[x(t^k + dt)] < 0.$$

Однако в качестве t_2 брать t^k или $t^{k+1} = t^k + dt$ не следует, так как такой выбор не обеспечивает при использовании в дальнейшем $u^*(t)$ из (12) постоянства $\Phi[x(t)]$. Нужно, используя ту или иную интерполяцию значений $\Phi[x(t^k)]$, $\Phi[x(t^{k+1})]$, найти значение $t_2 \in [t^k, t^{k+1}]$, обеспечивающее $|\Phi[x(t_2)]| \leq \epsilon$. Разумеется, можно использовать и t^{k+1} в качестве t_2 , но при шаге dt , существенно меньшем, чем этого требует задача в целом. Что касается величины ϵ , то она легко оценивается: так как $t_3 - t_2 \approx 100$, а выбор $u^*(t)$ из (12) обеспечивает постоянство $\Phi[x(t)]$, то для выполнения условия $|\Phi[x(t)] - \alpha| \leq \delta$ при $t \in [t_2, t_3]$ следует получить $|\Phi[x(t_2)]| < \delta/(t_3 - t_2)$.

Решение задачи методом последовательной линеаризации [96]. Общая схема алгоритма подробно описана в §§ 19–21, здесь мы напомним ее лишь в общих чертах.

1. Вводилось формальное время $0 \leq t \leq 1$, система уравнений заменялась на $\dot{x} = Tf(x, u)$.

2. На интервале $[0, 1]$ вводилась равномерная сетка с шагом 0,01, задача решалась в классе кусочно постоянных

$$u(t) = u_{n+1/2} \quad \text{при} \quad t \in (t_n, t_{n+1}).$$

3. При заданном $u(t)$ интегрировалась система (1) с шагом $dt = 0,001$, ее решение напоминалось в узлах сетки. Одновременно с этим в серединах интервалов сетки вычислялись и запоминались элементы матриц f_x, f_u .

4. Анализировался график функции $\Phi[x(t)]$, и на множестве $M = \{t: \Phi[x(t)] \geq 0,9 \max_t \Phi[x(t)]\}$ размещалось K точек аппроксимации (разумеется, под «множеством» мы понимаем лишь конечное множество узлов сетки с шагом $\Delta t = 0,01$).

5. Вычислялись производные входящих в задачу функционалов, для чего сопряженная система интегрировалась $K+2$ раза. Интегрирование сопряженных систем проводилось менее точно, чем интегрирование (1).

6. Формировалась и решалась задача линейного программирования, определявшая одновременно вариацию управления $\delta u(t)$ и вариацию параметра T .

По этой схеме были решены следующие задачи:

Задача 1. $\min x^3(T) R_3$ при условиях: $\Phi[x(t)] \leq \alpha$, $x^1(T) = R_3$, $\alpha = 7,65$, начальные данные a). Это та же задача, которая решалась в [19], и в качестве исходного управления бралась та же самая функция $u(t) = -0,2$, $K \leq 5$.

Задача 2. Она отличалась от первой следующим: начальные данные b), $\alpha = 5,9$, исходное $u(t) = 0,1$.

Задача 3. Начальные данные b), решалась задача на $\min_{u, T} \max_t \Phi[x(t)]$ при условиях $x^3(T) = 0,09308$; $x^1(T) = R_3$. Исходное $u(t) = 0,1$ ($x^3(T) R_3 = 600$).

Поясним некоторые технологические моменты.

1. Вариация управления $\delta u(t)$ искалась в области $\delta U(t): s^-(t) \leq \delta u(t) \leq s^+(t)$,

$$s^-(t) = \max \{-S, -0,2 - u(t)\},$$

$$s^+(t) = \min \{S, 0,2 - u(t)\}.$$

Здесь S — шаг процесса. В самом начале он задавался величиной $\sim 0,016 \div 0,02$, обеспечивающей $\sim 10\%$ точности линейного приближения (см. § 20), затем регулировался в зависимости от результатов вычислений. Наиболее жесткие требования к шагу предъявляет используемая здесь аппроксимация вариации дифференцируемого лишь по Гато функционала $F_0[u(\cdot), T] \equiv \max_t \Phi[x(t)]$. Точность условия $F_0 \leq \alpha$, как это следует

из теоремы (21.1), обеспечивается двумя факторами: числом точек аппроксимации K и малостью шага S , причем нежелательны как увеличение K (трудоемкость итерации), так и уменьшение S (замедление процесса минимизации). В начале было $K=1$, затем оно постепенно увеличивалось до заданного значения K^* . Было принято требование выполнения условия $F_0 \leq 1,005 \alpha$. Если в процессе решения при заданных значениях S и K оказывалось $F_0 > 1,005 \alpha$, то при $K < K^*$ происходило увеличение K на 1, а при $K = K^*$ — уменьшение S (умножением на 0,8). Однако этот алгоритм регулирования включался не сразу.

Дело в том, что исходное управление $\{u(\cdot), T\}$ не удовлетворяет дополнительным условиям задачи, поэтому первый этап решения — это решение терминальной задачи: ищется управление (с каким угодно значением минимизируемого функционала), удовлетворяющее дополнительным условиям (для задачи 3, например, эти условия: $x^3(T) = 0,09308$, $x^1(T) = R_3$). Признаком того, что решается терминальная задача, является отсутствие решения в задаче линейного программирования. Этот этап осуществляется с постоянным шагом S . После того как впервые встретится ситуация, в которой задача линейного программирования имеет решение, начинается собственно процесс минимизации и включается алгоритм регулирования S , K .

2. Заслуживает пояснения алгоритм размещения точек аппроксимации. В задачах 1 и 2 использовался следующий простой способ: выделялось множество узлов сетки $M = \{t: \Phi[x(t)] > 0,9F_0\}$, оно разбивалось на K примерно равных (по числу точек) частей, на каждой части находилась точка максимума $\Phi[x(t)]$; эта точка и становилась одной из K точек аппроксимации. Этот

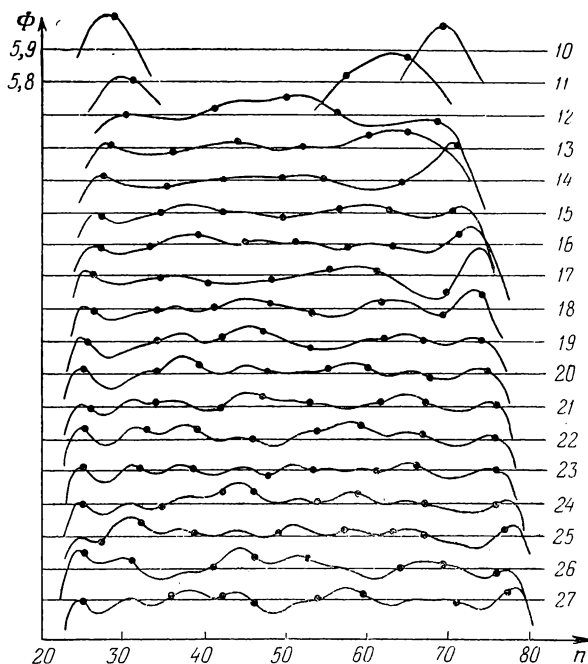


Рис. 48.

алгоритм в принципе приемлем (он, в частности, использовался и в задаче § 35). Однако ему присущ некоторый недостаток: если точка максимума попадает на границу двух частей, то соответствующие этим частям точки аппроксимации стягиваются в границе; они дублируют друг друга, и в то же время другие участки обнажаются. Теорема (21.1) показывает, что желательны следующие свойства размещения точек аппроксимации:

1) они должны по возможности равномерно покрывать множество M ;

2) точки с наибольшими значениями $\Phi[x(t)]$ должны входить в число точек аппроксимации или находиться возможно ближе от них.

Поэтому был добавлен алгоритм «расталкивания» таких почти слипшихся точек аппроксимации. Здесь он не рассматривается, так как его нельзя считать вполне удовлетворительным. Рис. 48 иллюстрирует работу этого алгоритма при решении задачи 2 ($K^*=8$). На этом рисунке показаны части функции $\Phi[x(t)]$ на разных итерациях (от 10-й до 27-й). Для наглядности каждая последующая кривая опущена на 0,1, а масштаб взят таким, чтобы были заметны величины $\sim 0,01$. Показаны и точки аппроксимации. Хорошо видны отмеченные выше дефекты их размещения. Например, на 27-й итерации излишне сближены точки 2, 3, 4 и точки 5, 6; в то же время образовались «пустоты» между 1-й и 2-й точками, между 6-й и 7-й. Эти дефекты хорошо заметны «на глаз» и легко исправляются «от руки», однако алгоритмизация подобных интуитивно тривиальных решений встречает затруднения. Во всяком случае, несколько казавшихся вполне разумными алгоритмов были отвергнуты после экспериментов. На рис. 49 функция $\Phi[x(t)]$ изображена полностью. В этом масштабе нарушение

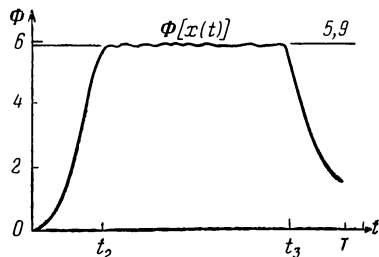


Рис. 49.

Таблица 2

Расчет 1						Расчет 2			
ν	$x^1(T)$	F_0	F^*	$x^3(T)R_3$	T	ν	F_0	F^*	$x^3(T)R_3$
0	6395,5	27,75	7299	186,3	140,0	0	27,75	7299	186,3
4	6399,4	17,40	2567	318,3	153,0	4	17,36	2630	302,4
8	6400,1	12,05	920	370,1	168,9	8	13,62	1233	332,1
12	6399,4	9,823	195	447,0	186,4	12	9,506	141,9	446,7
16	6398,0	8,288	13,8	504,3	205,8	15	8,366	20,8	493,2
19	6396,0	7,674	0,015	533,9	219,8	16	8,295	11,2	513,3
20	6396,0	7,699	0,039	527,2	219,1	17	8,104	3,94	521,3
22	6396,0	7,692	0,020	524,5	218,9	18	7,794	0,16	532,0
23	6396,0	7,847	0,039	521,6	218,6	19	7,690	0,006	530,0
24	6396,0	7,674	0,04	524,4	218,9	20	7,674	0,006	527,5
25	6396,0	7,770	0,150	521,5	218,5	21	7,660	0,007	526,1
26	6396,0	7,736	0,056	521,6	218,7	22	7,677	0,007	524,0
27	6396,0	7,746	0,124	521,8	218,8	23	7,740	0,097	522,3
28	6396,0	7,748	0,058	520,5	218,9	24	7,753	0,072	521,5
30	6396,0	7,779	0,016	520,4	218,0	25	7,697	0,016	522,8
32	6396,0	7,706	0,026	520,9	218,5	26	7,700	0,026	521,9
34	6396,0	7,715	0,050	520,3	219,0	27	7,711	0,042	521,8

условия $\Phi[x(t)] \leq \alpha$ почти незаметно. Процесс решения первой задачи показан в табл. 2 (расчет 1) следующими величинами: v , $x^1(T)$, $F_0 = \max \Phi[x(t)]$, F^* (та же величина, что и в табл. 1), $x^3(T)$ R , T . Минимальное значение $x^3(T)$ $R_3 = 520$. В [19] результат лучше, $x^3(T)$ $R_3 = 500$. Попытки продолжить расчет и получить дальнейшее уменьшение $x^3(T)$ R_3 ни к чему не привели. Именно это и заставило автора составить программу расчета точного решения и проконтролировать свои результаты. Они подтвердились. К сожалению, проконтролировать таким же образом результаты [19] не удалось, так как там не приведены значения $x(0)$. Вместо этого сообщается величина r_* (высота условного перигея), позволяющая после решения нелинейного уравнения восстановить $x(0)$. Видимо, из-за неточного решения этого уравнения наши начальные данные расходятся с используемыми в [19]. Расхождение несущественное, но затрудняющее сопоставление результатов. Сравним результаты решения одной и той же задачи разными методами.

Начальное приближение взято таким, что дополнительные условия грубо нарушены и первый этап (10—12 итераций) приводит к решению с относительно небольшим их нарушением. Оба метода на этом этапе показывают примерно одинаковую эффективность. Затем следует собственно оптимизация. В наших расчетах на 19-й итерации получено решение, сравнимое по величинам F^* и ΔL с решением, полученным методом штрафных функций на 150-й итерации, на 20-й итерации результат лучше, чем на 270-й, на 23—24-й — лучше, чем на 360-й в методе штрафных функций.

Разумеется, объективное сравнение методов требует учитывать не только число итераций, но и их временную цену. В наших расчетах основные затраты машинного времени для одной итерации связаны со следующими вычислениями:

- 1) интегрирование системы (1);
- 2) $(K+2)$ -кратное интегрирование сопряженной системы;
- 3) решение задачи линейного программирования. В методе штрафных функций основное время идет на
 - 1) интегрирование системы (1);
 - 2) однократное интегрирование сопряженной системы;
 - 3) r -кратное интегрирование прямой системы, связанное с выбором шага Δ (~ 12 —15 сек. на итерацию на БЭСМ-6).

Так как сопряженная система в наших расчетах интегрировалась менее точно, чем прямая (например, с шагом $dt = 1/400$), а наиболее трудоемкий элемент этого интегрирования — вычисление матриц f_x и f_u — производится один раз для всех сопряженных систем, то $(K+2)$ -кратное интегрирование по затратам времени мало отличается от однократного. В среднем временная цена одной итерации равна примерно трехкратному интегри-

рованию прямой системы. Если в методе штрафных функций проб для подбора Δ не производить ($r=0$), то временная цена итерации будет, видимо, в 1,5—2 раза больше времени интегрирования прямой системы (4). Если же $r=2, 3$, то временная цена итерации М. Ш. Ф. уже превосходит цену итерации М. П. Л. Таким образом, метод последовательной линеаризации оказался по крайней мере в 10—15 раз эффективнее метода штрафных функций. Другим его преимуществом является возможность автоматического решения задачи — все представленные здесь расчеты были произведены без вмешательства человека в процесс поиска. Наконец, последнее 12 итераций связано с уточнением решения. Хотя методом штрафных функций такую точность получить едва ли удастся, мы не будем относить это к преимуществам нашего метода. Дело в том, что содержательная ценность подобной точности не очень велика, хотя сам факт ее достижения свидетельствует о надежности и эффективности метода. Возникает вопрос — обязательно ли в М. Ш. Ф. производить достаточно трудоемкий подбор шага Δ . Нельзя ли, как это делается в наших расчетах, регулировать величину Δ без дополнительных интегрирований системы (1)? Видимо, без подбора Δ М. Ш. Ф. окажется еще менее эффективным. Для М. Ш. Ф. характерно использование конструкций функционалов типа (6), содержащих большой параметр (μ), что связано с очень плохими дифференциальными свойствами функционала $F_\mu[u(\cdot)]$: малое изменение $u(\cdot)$ сопровождается очень большим изменением функциональной производной, например. Поэтому информация, полученная на одной итерации, имеет сравнительно небольшую ценность на следующей, хотя траектория почти не изменилась. Таковы неизбежные последствия введения в задачу большого параметра: аналитическая формулировка задачи упрощается, но дифференциальные свойства входящих в задачу функций резко ухудшаются. В табл. 3 (расчет 3) показан процесс решения задачи 2. Эта задача несколько труднее с вычислительной точки зрения, так как интервал особого режима ($|u(t)| < 0,2$ при $t \in [t_2, t_3]$) в ее решении занимает отрезок времени $0,55 T \approx 135$. (В решении первой задачи этот интервал равен примерно 100). Поэтому было увеличено максимальное возможное число точек аппроксимации ($K^*=8$). В остальном решение этой задачи протекает аналогично решению первой.

Решение задачи методом проекции градиента. Задачи 1 и 2 были решены методом проекции градиента, подробно описанным в § 18. Схема вычислений в этом случае в основном совпадает со схемой вычислений методом последовательной линеаризации. Основное отличие в том, что вариация управления находится решением задачи квадратичного программирования. Задачи решались при тех же управляющих функциях и при тех же значениях входящих в них параметров, что и

Таблица 3

Расчет 3						Расчет 4			
ν	$x^1(T)$	F_0	F^*	$x^3(T)R_3$	T	ν	F_0	F^*	$x^3(T)R_3$
0	6443,7	6,440	3,27	900,9	240,0	0	6,440	3,27	900,9
4	6407,6	6,442	5,24	675,3	264,9	4	6,213	1,667	585,3
8	6396,2	6,506	6,82	558,5	275,3	8	6,516	8,515	554,5
10	6396,2	5,927	0,0036	586,3	281,1	10	6,140	1,00	550,4
12	6395,9	5,934	0,012	539,2	276,5	12	5,937	0,024	508,4
16	6396,0	5,956	0,062	497,7	263,4	16	6,016	0,109	491,0
20	6396,0	5,952	0,038	488,5	260,0	20	5,949	0,042	479,2
22	6396,0	5,920	0,007	485,5	256,8	22	5,956	0,029	478,1
23	6396,0	5,930	0,015	483,3	255,6	23	5,963	0,043	476,5
24	6396,0	5,961	0,054	480,8	254,4	24	5,959	0,038	475,9
25	6396,0	5,935	0,021	480,4	253,8	25	5,965	0,064	475,2
27	6396,0	5,943	0,024	477,7	251,7	26	5,953	0,037	475,3
29	6396,0	5,943	0,023	475,6	250,0	27	5,949	0,037	473,8
30	6396,0	5,939	0,021	473,9	249,2	29	6,956	0,032	473,8
31	6396,0	5,930	0,020	473,6	248,6	32	5,947	0,036	474,0
32	6396,0	5,938	0,027	472,4	248,1	33	5,947	0,026	473,0
33	6396,0	5,944	0,025	471,7	247,7	35	5,948	0,045	472,0

в расчетах 1 и 3 (табл. 2, 3). Ход решения задач показан в таблицах 2 (расчет 2, задача 1) и 3 (расчет 4, задача 2). Величины $x^1(T)$ и T не представлены, так как они почти те же, что и в расчетах 1 и 3. На рис. 50, 51 показаны оптимальные управляющие

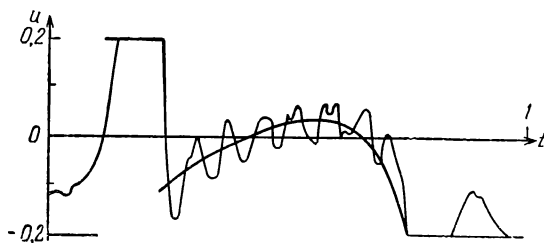


Рис. 50.

функции в задаче 2 — точная и найденные в расчетах 3, 4. Отметим следующие характерные моменты.

1. Заметное отличие численно найденных оптимальных $u(t)$ от точного на краях временного интервала. Это связано с тем, что значения $u(t)$ при $t \approx 0$ и при $t \approx 1$ не очень существенны и мало влияют на траекторию: при $t \approx 0$ движение происходит в вы-

соких слоях атмосферы, где плотность очень мала и движение аппарата «почти не управляемо»; при $t \simeq 1$ изменение управления, непосредственно влияющее лишь на величину ускорений (т. е. на \ddot{x}^2 и \ddot{x}^4), за оставшийся до конца полета небольшой отрезок времени не успевает сильно повлиять на координаты $x^1(1)$, $x^3(1)$, через которые выражаются функционалы F_1 и F_2 . Что касается функционала F_0 , то на его значение управление при $t \sim 1$ совсем не влияет.

2. На участке «особого режима» (t_2, t_3) численное управление отслеживает точное «в среднем», однако отклонения и здесь

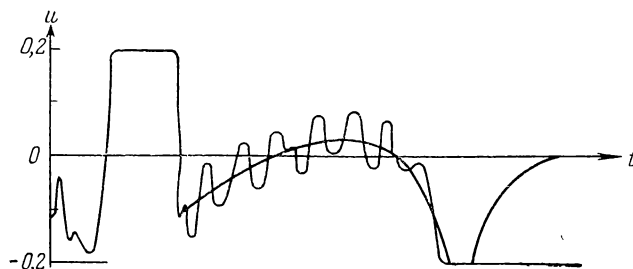


Рис. 51.

не очень малы. Это связано с некорректностью задачи оптимального управления (см. § 40) и возможностью сколь угодно точной аппроксимации особого решения функцией $u(t)$ релейного типа.

Решение задачи на $\min_{u(\cdot)} \max_t \Phi[x(t)]$. Метод последовательной линеаризации был использован для решения задачи 3, в которой минимизируется функционал F_0 , не имеющий производной Фреше. Параметры этого варианта задачи таковы, что оптимальная траектория содержит участок «особого режима» (t_2, t_3), на котором

$$\Phi[x(t)] = \max_t \Phi[x(t)] = F_0,$$

занимающий 62 счетных интервала из 100. В этом расчете был использован несколько иной алгоритм размещения точек аппроксимации, оказавшийся более удачным по сравнению с использованным в предыдущих расчетах. Он состоял в следующем.

1. Определялись множество узлов M_ϵ :

$$t \in M_\epsilon: \quad \Phi[x(t)] > F_0 - \epsilon, \quad \epsilon \simeq 0,2F_0$$

и число N_ϵ узлов t , попавших в M .

2. В качестве первой точки τ_1 аппроксимации назначалась точка максимума $\Phi[x(t)]$.

3. Из множества M_e исключались все узлы t , попавшие в некоторую окрестность узла τ_1 . Эта окрестность определялась условием типа

$$|t - \tau_1| \leq 0,6N_e/K,$$

где K — заданное число точек аппроксимации.

4. В качестве второй точки аппроксимации τ_2 бралась точка максимума $\Phi[x(t)]$ на оставшемся после исключения окрестности τ_1 множестве узлов M_e , затем из M_e исключалась еще и окрестность τ_2 и т. д.

Число точек аппроксимации K регулировалось следующим образом: пока решается терминальная задача, $K=1$. Затем, пока значение F_0 в процессе итераций понижается, K остается неизменным. Как только встретится ситуация, когда на очередном шаге вместо уменьшения F_0 его величина возрастает, K увеличивается на 1. Если K уже достигло предельного заданного значения, то вместо увеличения K уменьшается величина вариации управления. Однако в расчете, результаты которого приведены ниже, до этого дело не дошло, весь расчет проводится при $|\delta u(t)| \simeq 0,016$.

Таблица 4

ν	$x^1(T)$ предск.	$x^1(T)$ фактич.	$x^2(T)$ предск.	$x^2(T)$ факт.	F_0	K	T
0	—	6443,68	—	0,13980	6,44	1	240,0
1	6429,87	6431,35	0,11207	0,11222	6,91	1	228,0
2	6415,34	6415,61	0,1004	0,093101	7,47	1	229,4
3	6402,84	6403,84	0,0741	0,072546	8,17	1	251,4
4	6400,10	6400,90	0,07958	0,082313	7,47	1	263,9
5	6398,15	6399,10	0,09329	0,097061	6,90	1	277,1
6	6396,00	6396,31	0,093808	0,094495	7,21	2	277,9
7	6396,00	6395,82	0,093808	0,092830	6,76	2	284,5
8	6396,00	6396,01	0,093808	0,093780	6,48	2	282,2
9	6396,00	6396,06	0,093808	0,093967	6,23	2	281,0
10	6396,00	6396,06	0,093808	0,093991	6,01	2	279,6
11	6396,00	6396,05	0,093808	0,093982	5,81	2	277,7
12	6396,00	6396,05	0,093808	0,093988	5,68	2	275,6
13	6396,00	6396,04	0,093808	0,094002	5,79	3	273,1
14	6396,00	6396,02	0,093808	0,093870	5,51	3	274,2
15	6396,00	6396,04	0,093808	0,094070	5,42	3	274,2
16	6396,00	6396,03	0,093808	0,093990	5,37	3	273,6
17	6396,00	6396,05	0,093808	0,094073	5,59	4	272,9
18	6396,00	6396,02	0,093808	0,093916	5,34	4	271,5
19	6396,00	6396,01	0,093808	0,093899	5,291	4	271,1
20	6396,00	6396,01	0,093808	0,093922	5,294	5	270,6
21	6396,00	6396,01	0,093808	0,093872	5,36	6	269,1
22	6396,00	6395,99	0,093808	0,093739	5,28	6	270,2
23	6396,00	6396,01	0,093808	0,093928	5,25	6	269,7

В таблице 4 приведены следующие данные, характеризующие процесс решения задачи: ν — номер шага, значения координат $x^1(T)$ и $x^3(T)$, в терминах которых ставятся дополнительные условия ($x^1(T)=6396$, $x^3(T)=0,093808$, что соответствует дальности $x^3(T) R_3=600$). При этом приведены как фактические

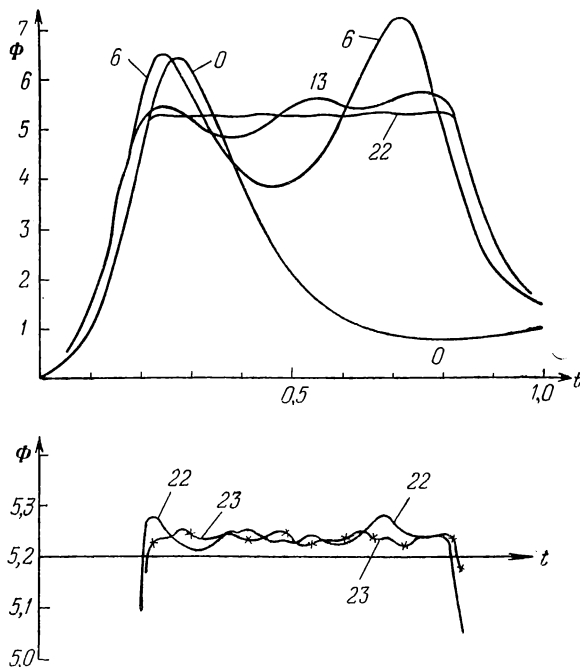


Рис. 52.

значения x^1 , x^3 , так и предсказанные на предыдущей итерации по формулам линейной теории возмущений:

$$F_0[u(\cdot) + \delta u(\cdot), T + \delta] = F_0[u(\cdot), T] + \int_0^T w(t) \delta u(t) dt + \frac{\partial F}{\partial T} \delta T.$$

Кроме того, в таблице приведены значения F_0 , K'_i и T .

На рис. 52 показана функция $\Phi[x(t)]$, полученная на итерациях с номерами $\nu=0, 6, 13, 22$. Кроме того, отдельно, в другом масштабе, показан участок, на котором $\Phi[x(t)] \simeq \max_t \Phi[x(t)]$ (для $\nu=22, 23$). В качестве исходного приближения бралась траектория с $u(t)=0,1$; первые шесть итераций составляют решение терминальной задачи, т. е. находится траектория, на которой выполнены дополнительные условия ($x^1(T)=639600$,

$x^3(T)=0,093808$). Только при определении $\delta u(t)$ на пятой итерации задача линейного программирования впервые в расчете имеет решение. Это видно из табл. 4: предсказанные значения $x^1(T)$, $x^3(T)$ начинают совпадать с заданными лишь с $v=6$. В этой задаче точное значение $\min F_0=5,23$; видно, что на 23-й итерации получено численное решение с ошибкой 0,4% в F_0 и 0,16% в $x^3(T)$. Полученное в этой задаче приближенное оптимальное управление отличается от точного примерно так же, как на рис. 50. Стоит еще пояснить соображения, используемые при назначении числа ϵ . Это число обычно достаточно велико. Основная неприятность, которой следует избегать и которая проявляется при слишком малом ϵ , состоит в том, что после вычисления $\delta u(t)$ и нового управления $u(t)+\delta u(t)$ на новой траектории $\max \Phi[x(t)]$ достигается в точке t^* , не входившей в множество M при вычислении δu . В этом случае процесс может приобрести характер «болтанки»: на одной итерации понижается значение Φ в точке t^* , но зато сильно повышается значение в другой точке t^{**} , не входившей в M и не контролировавшейся при вычислении δu . Следующая вариация $\delta u(t)$ вычисляется с множеством M , в которое входит t^{**} , но t^* не входит. Поэтому понижается $\Phi[x(t^{**})]$ за счет повышения $\Phi[x(t^*)]$ и т. д. Например, такая ситуация могла бы возникнуть после 6-й итерации (см. рис. 52), если при малом ϵ обе точки локальных экстремумов не войдут в M . Поэтому следует брать не очень малое ϵ , тем более, как нетрудно увидеть по рисункам 49, 52, увеличение ϵ до величины 1—2 мало влияет на M и еще меньше — на расположение точек аппроксимации.

Э к с п е р и м е н т. Были проведены расчеты с целью выяснить роль того основного элемента, который отличает используемый в расчетах алгоритм решения задачи квадратичного программирования от классического алгоритма строго выпуклого программирования. Речь идет о промежуточной минимизации $\|x\|$ (см. § 49). Для этого был проведен расчет по той же самой программе и в тех же условиях, что и расчет 4, с единственным изменением: в программе решения задачи квадратичного программирования был отключен блок минимизации $\|x\|$, т. е. эта задача решалась стандартным процессом строго выпуклого программирования, сходящимся, как известно, со скоростью геометрической прогрессии. Результат оказался следующим: затратив в 1,5 раза больше времени, чем этого потребовал весь расчет 4, удалось выполнить всего 6 итераций. К тому же эти 6 итераций относятся к самому легкому этапу решения задачи: варьируется взятая произвольно управляющая функция $u(t)=0,1$, дополнительные условия $F_0 \leq \alpha$ и $x^1(T)=R_3$ грубо нарушены; задача квадратичного программирования не имеет решения и при использовании § 49 это быстро выясняется. Кроме

того, на этом этапе использовались всего одна-две точки аппроксимации; таким образом, вертикальный размер задачи квадратичного программирования не превосходил 3. Этот пример показывает, как важен вычислительный эксперимент при разработке подобного рода вычислительных алгоритмов: ведь имеющиеся теоретические результаты не позволяют судить о фактической эффективности того или иного метода. Этот же пример показывает необходимость тщательной отработки даже на первый взгляд второстепенных деталей метода. Разумеется, если имеется в виду решать задачи, а не ограничиться указанием на возможность их решения.

§ 38. Вариационные задачи, связанные с проектированием ядерного реактора

Здесь будет описан опыт приближенного решения своеобразных задач оптимального управления. Они возникли при изучении возможностей оптимизации технических характеристик ядерного реактора за счет рационального размещения трех основных веществ, из которых составляется тело аппарата.

Математическая постановка задачи частично приведена в § 11. Напомним, что состояние управляемой системы определяется как первая собственная функция $x(t)$ линейного оператора

$$L(u)x = \lambda Q(u)x, \quad \Gamma x = 0. \quad (1)$$

Здесь $L(u)$ — линейный дифференциальный оператор, коэффициенты которого зависят от «управления» $u(t)$; $Q(u)$ — матрица $4 \rightarrow 4$, $x = \{x^1, x^2, x^3, x^4\}$. Конкретные выражения для L , Q и краевых условий $\Gamma x = 0$ см. в § 11. Мы будем придерживаться стандартного обозначения независимого аргумента буквой t , хотя в данной задаче это не время, а пространственная координата. Задача (1) рассматривается на интервале $[0, T]$. Управлением является трехмерная вектор-функция $u(t) = \{u_1, u_2, u_3\}$, определенная на интервале $[T_1, T_2] \subset [0, T]$, ее компоненты — концентрации в данной точке t трех характерных веществ: замедлителя (u_1), горючего (u_2) и поглотителя (u_3). Разумеется, коэффициенты $L(u)$ и $Q(u)$ зависят и от других компонент конструкции, но они в данной задаче не варьируются и явно в постановку задачи не входят. Правда, в связи с их наличием следовало бы, чтобы быть более аккуратным, использовать обозначения типа $Q(u, t)$, но мы этого не делаем ради простоты. Вычисление производных тех функционалов, в терминах которых будут ставиться различные задачи, разъяснено в § 11. Кроме того, в задаче имеются и геометрические ограничения допустимых значений компонент управления:

$$u \in U: \quad U_i^- \leq u_i \leq U_i^+, \quad i = 1, 2, 3. \quad (2)$$

Числа U_i^- , U_i^+ заданы, они определяются физическими и технологическими факторами. Была проведена серия расчетов оптимального размещения веществ с целью минимизации функционала

$$F_0[u(\cdot)] \equiv \frac{1}{x^2(0)} \max_{T_1 \leq t \leq T_2} \frac{u_2(t) x^2(t)}{u_1(t)}. \quad (3)$$

Заметим, что значение $F_0[u(\cdot)]$ не зависит от нормировки $x(\cdot)$. Содержательный смысл этого функционала связан с выделением теплоты на единицу вещества u_1 , имеющего наиболее низкую температуру плавления. В качестве дополнительного условия ставится следующее:

$$F_1[u(\cdot)] \equiv \lambda - 1 = 0. \quad (4)$$

Его содержательный смысл связан с требованием иметь дело с критическими системами. Напомним, что в (4) λ — крайняя точка спектра оператора (1). Если $u(\cdot)$ определяет оператор (1) с $\lambda > 1$, то соответствующая система подкритична и в ней ядерная реакция «затухает»; при $\lambda < 1$ система надкритична и «взрывается». Функционал $F_0[u(\cdot)]$ не имеет производной Фреше, так как в оптимальной (и в близких к ней) ситуации максимум в (3) достигается на интервалах, сравнимых с $[T_1, T_2]$. Кроме того, в выражение (3) явно входят компоненты управления. С этим связаны определенные трудности: аппроксимация приращения функционала формулой типа

$$\delta F_0[\delta u(\cdot)] = \max_{i=1, \dots, l} \{A_i \delta x^2(0) + B_i \delta x^2(\tau_i) + C_i \delta u_2(\tau_i) + D_i \delta u_1(\tau_i)\} \quad (5)$$

(где A_i , B_i , C_i , D_i вычисляются очевидным образом) оказывается неэффективной, так как $\delta u_1(t)$, $\delta u_2(t)$ суть произвольные функции. Поэтому задача была усложнена превращением компонент управления $u_1(t)$ и $u_2(t)$ в фазовые координаты. Вводились две новые компоненты управления $v_1(t)$ и $v_2(t)$, связанные с $u_1(t)$ и $u_2(t)$ уравнениями

$$\begin{aligned} \frac{du_1}{dt} &= v_1(t); & u_1(T_1) &= \alpha_1, \\ \frac{du_2}{dt} &= v_2(t); & u_2(T_1) &= \alpha_2, & T_1 \leq t \leq T_2. \end{aligned} \quad (6)$$

Теперь «управлением» в задаче является комплекс $\{v_1(\cdot), v_2(\cdot), u_3(\cdot), \alpha_1, \alpha_2\}$, а простые ограничения (2) превращаются в условия сформулированные в терминах не дифференцируемых по Фреше функционалов

$$\begin{aligned} F_2 &\equiv \max_t u_1(t) \leq U_1^+; & F_3 &\equiv \min_t u_1(t) \geq U_1^-, \\ F_5 &\equiv \max_t u_2(t) \leq U_2^+; & F_4 &\equiv \min_t u_2(t) \geq U_2^-. \end{aligned} \quad (7)$$

Задача решалась методом последовательной линеаризации (§§ 19—21). Напомним, что в соответствии с этой вычислительной схемой на $[T_1, T_2]$ вводилась сетка $T_1 = t_0 < t_1 < \dots < t_N = T_2$ и управляющие функции $u(t)$ (или $v(t)$) ищутся как кусочно постоянные:

$$u(t) = u_{n+1/2} \quad \text{при} \quad t_n < t < t_{n+1}, \quad n = 0, 1, \dots, N-1 \quad (N=85). \quad (8)$$

Можно было бы использовать аппроксимацию (5), взяв в качестве точек аппроксимации τ_i все точки t_i , в которых достигается максимум в (3). Однако $N=85$, и число таких точек, как мы увидим из дальнейшего, в некоторых расчетах было бы $\sim N$. Такой способ решения задачи привел бы к непосильным расчетам: ведь для вычисления $\delta F_0[\delta u(\cdot)]$ по (5) в виде функционала только от компонент $\delta u(\cdot)$ нужно было бы, используя стандартную технику исключения $\delta x(\tau_i)$ через интегралы от $\delta u(t)$, решать сопряженную систему l раз (l — число точек аппроксимации). Кроме того, определение вариации управления $\delta u(\cdot)$ было бы связано с решением задачи линейного программирования размером $3N \times l$, что при $l \sim N \sim 100$ было бы практически невозможно на той ЭВМ, которая использовалась в расчетах (примерно 50 000 операций в секунду, 4000 ячеек оперативной памяти).

После замены (6) аппроксимация (5) становится возможной при сравнительно небольшом значении l (в расчетах $l=3$). Правда, появляются «фазовые ограничения» (7), требующие аппроксимации типа (5). Однако вычисление производной Фреше для функционала $u(\tau_i)$ в высшей степени просто:

$$\delta u_1(\tau_i) = \delta \alpha_1 + \int_{T_1}^{\tau_i} \delta v(t) dt.$$

Общее число точек аппроксимации ограничений (7) было в расчетах равно 8. Они распределялись в зависимости от создавшейся в расчетах ситуации. Так, например, $u_1(t)$ обычно не достигало значения U_1^+ , и соответствующее условие в (7) игнорировалось.

Стандартная итерация, позволяющая от управления $\{u(\cdot)\}$ перейти к новому, лучшему $\{u(\cdot) + \delta u(\cdot)\}$, состояла из следующих вычислений.

1. С данным управлением $u(\cdot)$ решалась задача (4).

Это решение предполагало внутренний итерационный процесс: использовался популярный среди физиков метод «итераций источника». Дело в том, что краевая задача $Lx=f$, $\Gamma x=0$ при заданной правой части f легко решается прогонкой, а собственная функция и собственное число λ могут быть найдены быстро сходящимися итерациями (v — номер итерации):

$$Lx^{v+1} = \lambda^v Q x^v, \quad \Gamma x^{v+1} = 0, \quad v = 0, 1, \dots, \quad \lambda^{v+1} = \lambda^v \frac{\|x^v\|}{\|x^{v+1}\|}. \quad (9)$$

В качестве исходного приближения x^0 бралась функция $x(t)$, полученная на предшествующем этапе вычислений, так что было достаточно пяти итераций (9) (кроме самого начала процесса вариаций управления).

2. Решалось (таким же методом (9), но без пересчета λ) сопряженное уравнение

$$L^*\psi = \lambda Q^*\psi; \quad \Gamma^*\psi = 0. \quad (10)$$

3. Трижды (так как $l=3$) решались уравнения

$$L^*\psi^i - \lambda Q^*\psi^i = Y^i[t]; \quad \Gamma^*\psi^i = 0; \quad i = 1, \dots, l. \quad (11)$$

Решения этих уравнений позволяют в (5) выражения в фигурных скобках переписать в терминах только функций $\delta u(t)$. Определение $\psi^i(t)$ из (11) осуществлялось примерно такими же итерациями (9).

4. От выражений для вариаций в терминах $\delta u(t)$ нетрудно перейти к эквивалентным в терминах вариаций управляющего комплекса $\{\delta v_1(\cdot), \delta v_2(\cdot), \delta u_3(\cdot), \delta \alpha_1, \delta \alpha_2\}$.

5. Далее формируется и решается задача линейного программирования, в которой $\sim 3 \times 85$ неизвестных и не более 12 условий. Решение этой задачи определяет вариацию управления.

В целом, если Δt — машинное время, необходимое для решения задачи (1) методом (9), то время всей итерации равно примерно $6\Delta t$: Δt на решение (1), Δt на решение (10), $3\Delta t$ на (11) и Δt на все остальное. Заметим, что разностная схема, аппроксимирующая уравнения (1), (10), (11), имела шаг, меньший шага сетки $t_{n+1} - t_n$ для управления (8), и содержала 400—500 точек, чем обеспечивалась необходимая точность. Формулы, определяющие зависимость L и Q от u , здесь не приводятся, так как они хотя и не очень сложны, но все же громоздки и для дальнейшего не очень существенны. Они приведены в препринте [36]. Среднее время тех расчетов, которые будут представлены, колебалось от 20 до 40 минут. Оно могло бы быть существенно уменьшено за счет сокращения, например, числа узлов сетки N до 20—30.

На рис. 53 представлены результаты численного решения задачи, когда варьировалась только одна компонента управления: $u_1(t)$ или $u_2(t)$. Эти задачи, кроме всего прочего, интересны еще и тем, что в них по результатам расчетов удалось угадать структуру точного решения. Зная эту структуру, можно было построить алгоритм вычисления точного решения (разумеется, тоже приближенный, но имеющий много меньшую ошибку, чем алгоритм поиска). Упомянутая структура точного решения определялась следующим, угаданным по численным результатам, свойством:

$$\frac{1}{x^2(0)} \frac{u_2(t) x^2(t)}{u_1(t)} = \max_{t_1 \leq t \leq t_2} \left\{ \frac{1}{x^2(0)} \frac{u_2(t) x^2(t)}{u_1(t)} \right\} \text{ при всех } t \in [T_1, T_2]. \quad (12)$$

На рис. 53, а приведены результаты решения задачи с вариацией только u_2 ; $u_1(t)$ и $u_3(t)$ — фиксированы. Показаны: функция $u_2(t)$ — точная (сплошная линия) и полученная процессом поиска (отмечена «X»), и функция $F(t) = \frac{1}{x^2(0)} \frac{x^2(t) u_2(t)}{u_1(t)}$ — точная (пунк-

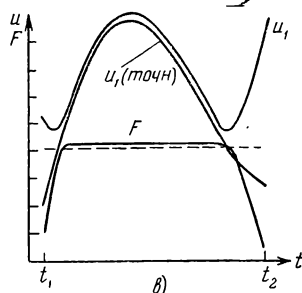
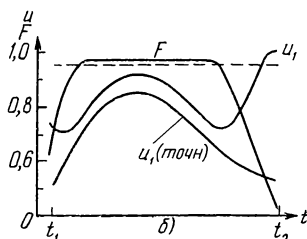
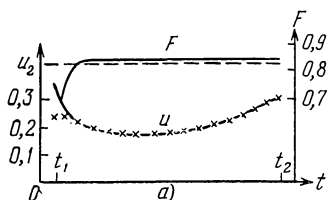


Рис. 53.

тир) и приближенная (сплошная линия). Значение функции $F_0 = 0,94$ вместо точного значения $F_0 = 0,93$ (ошибка $\sim 1\%$). Условие $\lambda = 1$ выполнено с точностью до $0,002$, ограничения $U_2^- \leq u_2 \leq U_2^+$ не работают, так как всюду $U_2^- < u_2(t) < U_2^+$. На рис. 53, б и в приведены аналогичные кривые для задачи, в которой варьировалось только $u_1(t)$ при фиксированных $u_2(t)$, $u_3(t)$.

Угадать по численным результатам структуру (12) несложно.

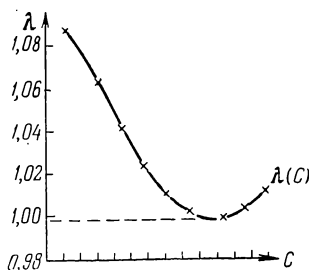


Рис. 54.

Поясним теперь алгоритм вычисления точного решения. Он существенно опирается на структуру (12) и на то, что варьируется лишь одна компонента управления. Задавалось некоторое число C и какая-нибудь функция $u_2(t)$. С этим u решалась задача (1), и определялись $x(t)$ и λ . Далее $u_2(t)$ пересчитывалось из уравнения

$$\frac{1}{x^2(0)} \frac{u_2(t) x^2(t)}{u_1(t)} = C, \quad \text{т. е.} \quad u_2(t) = C \frac{u_1(t) x^2(0)}{x^2(t)}. \quad (13)$$

С новым $u_2(t)$ решалось (1), снова пересчитывалась функция $u_2(t)$ и т. д. до установления. Процесс оказался быстро сходящимся; разумеется, назначение исходных C и $u_2(t)$ опиралось на результаты приближенного решения задачи. Этот процесс, таким образом, определял числом C значение $\lambda(C)$ и функцию $u_2(t, C)$. Затем обычным способом решалось уравнение $\lambda(C) = 1$.

При решении задачи с $u_1(t)$ (рис. 53, б) расхождение между точным оптимальным $u_1(t)$ и найденным численно оказалось относительно большим — хотя по значению F_0 точность приближенного решения $\sim 1\%$ (точный $\min F_0 = 0,95$, приближенное решение дает $F_0 = 0,96$). Было интересно выяснить, с чем это связано. На рис. 54 показана функция $\lambda(C)$, построенная по нескольким точкам C с помощью итерационного процесса (13). Видно, что уравнение $\lambda(C) = \lambda_0$ имеет два решения при $\lambda_0 \geq 0,997$. Уравнение $\lambda(C) = 0,996$ решения не имеет. Таким образом, при $\lambda(C) = 1$ мы находимся так сказать на границе существования оптимального решения вида (12)*. Если бы дополнительное условие имело вид $\lambda = \lambda_0 < 0,996$, то решения вида (12) уже, наверное, не было бы. Оптимальное решение имело бы какую-то другую структуру. Видимо, $u_1(t)$ на рис. 53, б и несет на себе следы этой другой структуры. Для проверки этого предположения было проведено решение вариационной задачи с варьированием только $u_1(t)$ при условии $\lambda = 1,035$. Это решение (и соответствующие этой задаче точные функции) изображены на рис. 53, в. Видно, что совпадение $u_1(t)$ с точным стало намного лучше. Было бы интересно получить численное решение и при λ , допустим, 0,96, когда структура (12) неосуществима. К сожалению, эта мысль пришла автору тогда, когда машина, на которой проводились расчеты, была демонтирована как устаревшая, а программа, написанная в кодах, оказалась, таким образом, утраченной (описываемые здесь расчеты проводились в 1965—1967 гг.). Предположения, которые были сделаны в связи с решениями задач (рис. 53), не очень строги: точно так же, сходимость алгоритма (13) не гарантирована. Все это было подробно описано как типичный пример тех средств, к которым часто обращается вычислитель, имеющий дело с достаточно сложной прикладной задачей. Успех является оправданием применяемых средств.

Т а б л и ц а 1

Задача I					Задача II		Задача IV			
ν	F_0	λ	$\min_t u_1(t)$	$\max_t u_2(t)$	ν	F_0	ν	F_0	λ	λ'
0	4,403	1,005	0,875	0,200	0	2,211	0	0,801	1,030	1,000
3	3,634	0,998	0,765	0,260	3	2,054	3	0,850	1,002	0,994
6	3,073	1,003	0,695	0,314	6	1,893	6	0,845	1,004	1,000
9	2,782	1,003	0,695	0,368	12	1,762	9	0,798	0,9996	1,005
12	2,569	1,001	0,699	0,422	18	1,700	15	0,716	0,9986	1,002
15	2,355	1,005	0,688	0,492	24	1,659	21	0,701	0,9991	1,004
18	2,151	0,995	0,692	0,572	30	1,641	27	0,673	0,997	0,996
21	2,063	1,003	0,693	0,601	36	1,627	33	0,645	1,002	1,001
24	2,040	1,000	0,695	0,604	42	1,610	39	0,625	0,997	1,002
27	2,026	0,997	0,692	0,605	48	1,602	45	0,618	0,995	0,999
30	2,019	1,003	0,694	0,606	54	1,598	54	0,569	0,997	0,997
					57	1,597	60	0,565	0,999	1,001
					60	1,597	66	0,556	1,006	1,000
					63	1,594	72	0,546	1,002	0,999
					66	1,592	75	0,543	1,000	1,001
					69	1,592	78	0,545	0,9998	1,001

*) Уравнение $\lambda(C) = 1$ имеет два решения. Было найдено и управление, соответствующее второму корню. Оно оказалось существенно худшим по значению F_0 .

В следующей задаче варьировались все три компонента управления $u_1, u_2, u_3(t)$, причем $u_1(t)$ вышло на свою нижнюю границу $U_1^- = 0,7$ на всем интервале $[T_1, T_2]$, а $u_2(t)$ — на границу $U_2^+ = 0,6$ на интервале длиной $0,4$ ($T_2 - T_1$). Оба ограничения аппроксимировались по трем точкам. В табл. 1 (расчет I) подробно показан ход решения задачи. Приведены следующие величины: номер итерации ν и значения $F_0, \lambda, \min_t u_1(t), \max_t u_2(t)$ на данной итерации.

Здесь точное решение угадать не удалось, и доверие к этим результатам основано, прежде всего, на опыте успешного решения близких по содержанию задач (рис. 53). На рис. 55 процесс решения этой задачи показан эволюцией функций $\Phi[t], u_1(t), u_2(t)$ (Φ — функция, максимум которой определяет значение $F_0[u(\cdot)]$). Отметим следующие обстоятельства. Прежде всего, ограничение $u_1 \geq 0,7$ — существенно, его нарушение очень выгодно с точки зрения минимизации F_0 . Это следует из того, как стремительно функция $u_1(t)$ выходит на это ограничение (на 6-й, примерно, итерации). Выход $u_1(t)$ на границу $0,7$ приводит к тому, что в дальнейшем $u_1(t)$, по существу, не меняется. Кстати, и темп падения F_0 в процессе итераций после этого заметно упал. Второе значительное падение темпа изменения F_0 произошло после выхода $u_2(t)$ на

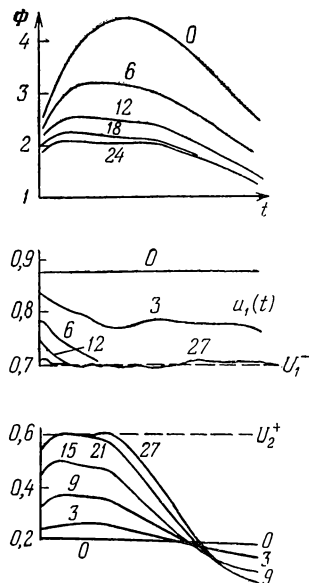


Рис. 55.

верхнюю границу $0,6$ ($\nu \approx 19$). Поучительной оказалась третья задача, в которой, как и в первой, варьировалась только компонента $u_2(t)$. Однако, фиксированная компонента $u_1(t)$ была задана разрывной, в связи с чем и в точном решении, как в дальнейшем выяснилось, должен был образоваться разрыв в $u_2(t)$ (в той же точке t , где был разрыв в $u_1(t)$). Замена $\dot{u}_2(t) = v_2(t)$ привела здесь к затруднениям, о которых уже говорилось: для образования разрыва в $u_2(t)$ нужно процессом малых вариаций $v(t) \rightarrow v(t) + \delta v(t)$ построить в $v(t)$ δ -функцию. Этого, в сущности, не получилось, точнее, разрыв оказался сильно размазанным. Процесс поиска минимума F_0 стабилизировался при значении $F_0 = 1,7$. Эта стабилизация произошла в ситуации, сравнительно далекой от оптимальной. Причина ее — неточность аппроксимации (5) при $l=3$. На рис. 56, а изображена точная оптимальная функция $u_2(t)$ (сплошная линия) и приближенная (нанесена «0»).

(Результаты и здесь позволили предположить структуру (12) точного решения и найти его итерациями (13)). Видно, что есть участки заметного отклонения приближенного $u_2(t)$ от точного. При вычислении вариации управления $\delta u_2(t)$ мы имеем два ресурса понижения F_0 : истинный ресурс, связанный с отличием $u_2(t)$ от точного, и ложный, связанный с неточностью аппроксимации (5) и позволяющий понижать значения $\Phi[t]$ в точках аппроксимации τ_1, τ_2, τ_3 за счет повышения значений в других точках, где $\Phi[t] \simeq \max_t \Phi[t] = F_0[u(\cdot)]$. Когда этот второй ресурс становится, так сказать, мощнее первого, процесс поиска приобретает характер «болтанки»: понижения F_0 фактически не происходит. В данном случае, ввиду сложности искомого решения (в терминах $v(t)$), это произошло сравнительно далеко от истинного $\min F_0 = 1,585$, т. е. численное решение содержало ошибку $\sim 7\%$. В связи с этим был предложен и успешно применен в данной задаче метод, позволяющий преодолеть трудности, порождаемые заменой $\dot{u} = v$. Именно, вариация $\delta u(t)$ ищется в виде

$$\delta u(t) = \delta u'(t) + \delta u''(t). \quad (14)$$

При этом $\delta u'(t)$ — вариация, малая не только сама, но имеющая еще и малую производную. Конкретно, $\delta u'(t)$ определяется уравнением

$$\frac{d\delta u'}{dt} = \delta v(t); \quad \delta u'(T_1) = \delta \alpha, \quad (15)$$

где $\delta v(t)$ — произвольная малая функция (разумеется, имеется в виду сеточная функция типа (8)).

Что касается $\delta u''(t)$, то это произвольная малая сеточная функция типа (8), однако она отлична от нуля только в тех точках t отрезка $[T_1, T_2]$, где

$$\Phi[x(t), u(t)] \leq 0,95F_0[u(\cdot)] \quad (F_0 \equiv \max_t \Phi[x(t), u(t)]). \quad (16)$$

Таким образом, введение компоненты $\delta u''(t)$ не портит аппроксимации (5) и, в то же время, позволяет получать разрывные $u(t)$ без δ -функций. Процесс решения третьей задачи с использованием приема (14) показан в табл. 1 (задача II) эволюцией в процессе поиска величины F_0 . Значение λ при этом совпадает с 1 так же, как в остальных случаях, и не приводится. Рис. 56 иллюстрирует результаты численного решения задачи. На рис. 56, а показаны оптимальные функции $u_2(t)$ — точная (сплошная линия), приближенная, полученная без применения (14), («0») и приближенная, полученная с помощью (14), («×»). На рис. 56, б показаны функции $\Phi[x(t), u(t)]$, полученные с применением (14) ($\max_t \Phi[x(t), u(t)] = 1,592$) и без этого приема ($\max_t \Phi[x(t), u(t)] = 1,7$).

На рис. 56, *в* и 56, *г* показана эволюция функций $u_2(t)$ и $\Phi[t]$ в процессе поиска.

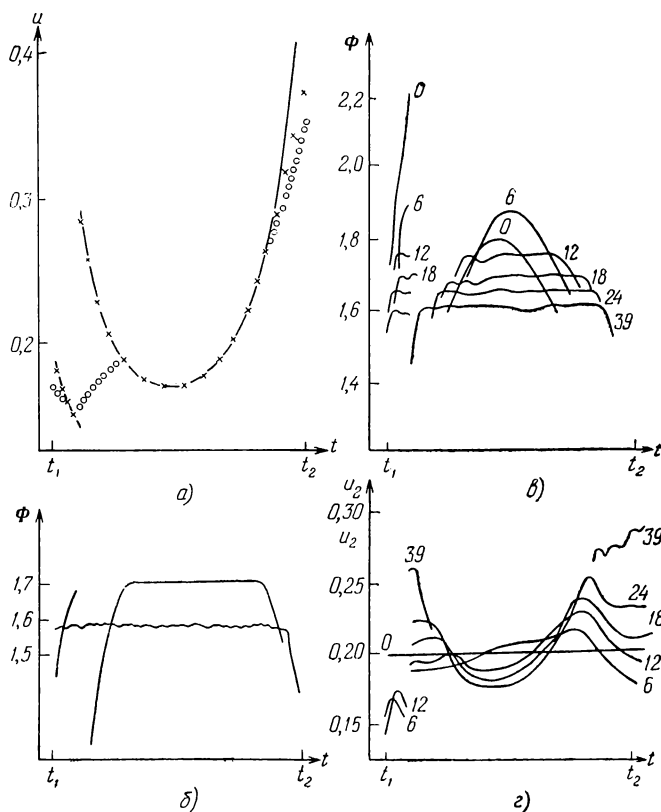


Рис. 56.

Четвертая задача, в которой варьировались все три компоненты $u(t)$, была осложнена еще одним дополнительным условием $\lambda' = 1$, где λ' — крайняя точка спектра оператора типа (1), но с другими значениями входящих в него коэффициентов, зависящих от того же самого управления $u(\cdot)$. Процесс решения этой задачи показан в табл. 1 (задача IV) эволюцией величин $F_0[u(\cdot)]$, λ и λ' . На рис. 57 показано изменение функции $\Phi[x(t), u(t)]$ в процессе поиска. В этой задаче плато в функ-

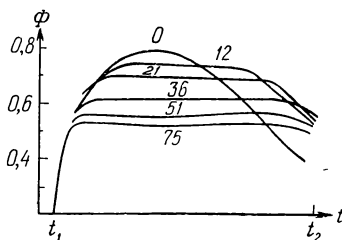


Рис. 57.

ции образовалось довольно рано (при $v \approx 10-12$), и после этого произошло значительное уменьшение F_0 (от $F_0=0,8$ до $F_0=0,545$). Этим, а также наличием дополнительного условия $\lambda'=1$, определяется сложность расчета.

§ 39. Об одном способе аппроксимации не дифференцируемого функционала

Построение минимизирующей последовательности управлений обычно так или иначе использует производные от входящих в задачу функционалов. Поэтому значительные трудности связаны с решением тех задач, в постановку которых входят функционалы, не имеющие производной Фреше, но все же дифференцируемые в смысле Гато (по направлениям). К таким функционалам приводят обычно конструкции типа

$$F[u(\cdot)] \equiv \max_t \Phi[x(t)]. \quad (1)$$

Легко понять популярность идеи аппроксимации такого функционала другим, имеющим уже производную Фреше, тем более, что и способы аппроксимации технически несложны, и появляется соблазнительная возможность доказать теорему о том, что подобную аппроксимацию можно сделать сколь угодно точной. В частности, нередко (см., например, [27]) предлагается аппроксимировать (1) дифференцируемым функционалом

$$F^{(p)}[u(\cdot)] \equiv \left\{ \int_0^T \Phi^p[x(t)] dt \right\}^{1/p} \quad \left(\lim_{p \rightarrow \infty} F^{(p)}[u(\cdot)] = F[u(\cdot)] \right). \quad (2)$$

К сожалению, предложения подобного рода намного опережают опыт их фактического использования. Во всяком случае, автору неизвестны публикации, в которых сообщалось бы об использовании аппроксимации (2), и о том, что из этого вышло. Но очевидно, что аппроксимация (2) обладает дефектом, характерным для многих подобных конструкций, содержащих большой параметр (для метода штрафных функций, например): при малых p аппроксимация неточна, при больших p она точна, но функционал $F^{(p)}[u(\cdot)]$ плохо линеаризуется: это означает, что формула

$$\delta F^{(p)}[\delta u(\cdot)] = \left(\frac{\partial F^{(p)}}{\partial u(\cdot)}, \delta u(\cdot) \right)$$

имеет точность (скажем, $\sim 10\%$) лишь при очень малых δu , что в свою очередь заставляет вести процесс построения минимизирующей последовательности управлений слишком малым шагом.

Разумеется, все сказанное выше носит качественный характер, и более обоснованное суждение о практической ценности аппроксимации (2) можно вынести лишь на основании опыта. Автором был проведен вычислительный эксперимент на задаче о спуске космического аппарата. Эта задача и опыт ее решения подробно описаны в § 37. Был повторен расчет, который в § 37 отражен в табл. 3 и рисунках 49, 51 с одним лишь изменением: вместо условия $\max_i \Phi[x(t)] \leq 5,9$ (см. § 37) использовалось ограничение дифференцируемого функционала:

$$\kappa \left\{ \int_0^T \Phi^p[x(t)] dt \right\}^{1/p} \leq 5,9, \quad (3)$$

где κ — множитель, выбираемый на каждой траектории так, чтобы было выполнено соотношение $\kappa F^{(p)} = \max_t \Phi[x(t)]$. Если бы этот множитель не вводился, результаты были бы еще хуже (впрочем, ненамного, так как $\kappa \simeq 1,3$, как будет видно из расчетов). Заметим еще, что определенные предосторожности (при больших p) следует принять при вычислении $\int \Phi^p dt$. Оно осуществлялось так: находилось значение $F_0 = \max_t \Phi[x(t)]$ и затем

$$\left\{ \int_0^T \Phi^p dt \right\}^{1/p} = F_0 \left\{ \int_0^T \left[\frac{\Phi}{F_0} \right]^p dt \right\}^{1/p}.$$

Прежде чем обсуждать результаты эксперимента, которые автор считает отрицательными и ставящими под сомнение практическую ценность аппроксимации (2), следует сделать несколько оговорок.

1. Полученные результаты сравниваются с нашими расчетами в § 37. Сравнение с решением той же задачи методом штрафных функций, также подробно комментируемым в § 37, было бы более благоприятным, но не настолько, чтобы изменить отрицательное отношение к (2).

2. Автор неоднократно подчеркивал, что всякая идея требует еще и подходящего технологического оформления. К сожалению, в работах, предлагающих аппроксимацию (2), эти вопросы совершенно не разработаны, и автору пришлось взять это на себя. Скептическое отношение к подобным идеям могло, конечно, привести к тому, что не было сделано все возможное, чтобы довести идею до эффективного вычислительного алгоритма. Во всяком случае, те неудачные попытки использовать (2), которые были сделаны, показывают, что предложить аппроксимацию (2) — еще не значит создать метод решения задач с функционалами

типа (1). Это лишь возможный в принципе подход, из которого, может быть, удастся сделать эффективный метод (затратив усилия много большие тех, которые нужны для того, чтобы придумать нечто, подобное (2)), а может быть, ничего и не выйдет. Автор полагает, что, скорее всего, ничего не получится.

Тот, кто с этим не согласен, сможет при желании опровергнуть эту точку зрения, доведя идею аппроксимации (2) до расчета: задача в § 37 описана так, что допускает воспроизведение, а приведенные там результаты расчетов дают представление о достигнутой точности и вычислительной их цене.

3. Эксперимент проводился на сравнительно сложной задаче, в которой $\max_t \Phi[x(t)]$ достигался на большей части интервала управления $[0, T]$. Можно надеяться, что в более простых ситуациях аппроксимация (2) окажется более работоспособной. Однако, как будет видно из дальнейшего, трудности встретились в начальной стадии расчета, когда $\Phi[x(t)]$ имела всего две точки локальных максимумов.

Перейдем к результатам. Первый расчет проводился при $p=10$ и ограничении на вариацию управления $|\delta u(t)| \leq 0,016$, т. е. в тех же условиях и при той же начальной траектории, что и в соответствующем расчете § 37. Вариация управления проводилась методом проекции градиента (см. § 18). Заметим, что для перехода от исходного управления к оптимальному нужно изменить $u(t)$ на величину порядка 0,3—0,4. При шаге $\delta u \approx 0,016$ это в принципе можно сделать за ~ 25 шагов, что и наблюдается в наших расчетах в § 37. Результаты расчетов, использующих аппроксимацию (2), приведены в табл. 1, где показаны следующие величины: ν — номер итерации, значения $F^{(\nu)}$ и $F_0 = \max_t \Phi[x(t)]$. Величина

минимизируемого функционала в таблице не приведена, так как до него дело не дошло: ведь сначала должна быть решена терминальная задача; нужно найти управление, для которого $x^1(T) = 63,96$ и $\max_t \Phi \leq 5,9$. Расчет оказался совершенно неудачным,

процесс принял характер «болтанки»: на четных итерациях $\max_t \Phi \leq 5,9$, и вариация управления выбирается с целью понижения $x^1(T)$ (с 64,44 до 64,24). Однако расчет $\delta F^{(\nu)}$ по формулам линейной теории возмущений для $|\delta u(t)| \sim 0,016$ настолько неточен, что на следующей итерации $\max_t \Phi \sim 6,1$, хотя по линейной теории должно было бы быть $\max_t \Phi \leq 5,9$. На нечетных итерациях происходит понижение $\max_t \Phi$ за счет роста $x^1(T)$. Такой характер процесса поиска типичен для ситуаций, в которых шаг $|\delta u(t)|$ слишком велик и должен быть уменьшен.

Второй расчет проводился в тех же условиях, что и первый, но параметр S выбирался таким, чтобы шаг процесса $\delta u(t) \simeq 0,0016$. Результаты показаны в той же таблице теми же

Таблица 1

1-й расчет				2-й расчет				3-й расчет				
ν	$x^1(T)$	$F(p)$	F_0	ν	$x^1(T)$	$F(p)$	F_0	ν	$x^1(T)$	$F(p)$	F'_0	F''
0	64,437	8,69	6,44	0	64,437	8,692	6,44	0	64,437	7,35	6,44	
1	64,478	8,23	6,13	1	64,434	8,643	6,41	1	64,431	7,31	6,41	
2	64,522	7,83	5,84	2	64,430	8,595	6,38	2	64,427	7,27	7,38	
3	64,363	8,24	6,13	3	64,427	8,547	6,34	5	64,411	7,16	6,28	
4	64,437	7,84	5,85	4	64,423	8,500	6,31	10	64,389	6,98	6,13	
5	64,240	8,46	6,13	5	64,422	8,454	6,28	15	64,384	6,81	5,98	2,06
10	64,416	7,84	5,85	10	64,410	8,232	6,13	20	64,335	6,71	5,90	3,65
11	64,240	8,64	6,13	11	64,411	8,189	6,10	21	64,286	6,77	5,91	5,49
20	64,415	7,84	5,85	15	64,419	8,024	5,98	22	64,282	6,77	5,90	5,67
21	64,239	8,46	6,12	16	64,422	7,984	5,944	23	64,233	7,60	5,91	6,75
30	64,413	7,84	5,84	20	64,383	7,907	5,897	24	64,282	6,77	5,90	5,66
31	64,236	8,46	6,12	21	64,330	7,949	5,914	25	64,233	7,60	5,91	6,75
32	64,413	7,84	5,84	25	64,273	8,286	6,075	26	64,282	7,76	5,90	5,66
33	64,239	8,46	6,12	26	64,329	7,912	5,901	27	64,233	7,59	5,91	6,75
				27	64,273	8,280	6,064	28	64,282	6,76	5,90	5,65
				28	64,274	8,263	6,031	34	64,283	6,76	5,90	5,63
				34	64,330	7,912	5,901	35	64,234	7,58	5,92	6,73
				35	64,274	8,258	6,021					

величинами. Существенного продвижения вперед не получилось, поиск застрял примерно в той же ситуации, только амплитуда «болтанки» стала несколько меньшей.

Отметим еще два обстоятельства. В среднем время на одну итерацию в данном эксперименте в ~ 2 раза меньше, чем на итерацию в расчетах § 37, так что затраты машинного времени на 33—35 итераций составили $\sim 50\%$ от затрат на каждый из расчетов § 37. Но, с другой стороны, метод, использующий аппроксимацию (2), в первых двух расчетах «застрял» в сравнительно простой ситуации, когда функция $\Phi[x(t)]$ достигает максимума в единственной точке, и функционал $\max_t \Phi[x(t)]$ дифференцируем

по Фреше. В расчетах § 37 на этом этапе решения задачи (решение терминальной задачи) никаких осложнений не было. Если мы попытаемся идти стандартным путем уменьшения шага δu до величины, допустим, 0,00016, то, даже если предположить, что этим все проблемы будут решены, следует рассчитывать на ~ 2000 —3000 итераций для перехода от исходного управления до оптимального. Это неприемлемо.

Третий расчет проводился при $|\delta u| \sim 0,0016$, $p=20$. Здесь удалось продвинуться несколько дальше, после 22-й итерации функция $\Phi[x(t)]$ имеет уже две точки локального максимума с примерно равными значениями; одна из этих точек t_1 расположена вблизи левого конца интервала времени $[0, T]$, вторая — в точке $t_2=T$. Процесс поиска показан в таблице величинами v , $x^1(T)$, $F^{(p)}$, F' , F'' (последние две величины — значения $\Phi[x(t)]$ в точках локальных максимумов t_1 , t_2). И в этом случае процесс поиска «застрял» на стадии решения терминальной задачи, дальнейшее продвижение требует дальнейшего уменьшения шага поиска δu . То, что значение $\Phi[x(t_1)]$ с хорошей точностью держится около значения 5,9, объясняется следующим: на величину $x(t_1)$ влияет

только управление $u(t)$ при $0 \leq t \leq t_1$, и интегральный шаг $\int_0^{t_1} |\delta u| dt$

мал, он существенно меньше интегрального шага $\int_0^T |\delta u| dt$, влияющего на $x(t_2)$. Поэтому величина $F'' = \Phi[x(t_2)]$ колеблется в существенно больших пределах.

Одной из причин неудачи этого расчета могла быть вариация T : в конце третьего расчета $\delta T \approx 0,007 T$, причем изменение T носит характер колебания $T \rightarrow T \cdot 1,007$ на нечетных итерациях, $T \rightarrow T \cdot 0,993$ на четных. Поэтому было решено облегчить ситуацию, исключив T из числа варьируемых элементов управления.

Четвертый расчет проводился при $p=20$, $T=240$ (это близко к оптимальному T). Среднее значение $|\delta u|$ было равно $\sim 0,016$ на

Таблица 2

$p = 20$					
ν	$ \delta u(t) $ среднее	$x^1(T)$	$\bar{p}(p)$	F'	F''
0	0,016	6434,7	7,35	6,44	
1	0,016	6446,4	6,98	6,13	
2	0,016	6448,3	6,65	5,84	
3	0,016	6434,2	6,98	6,13	2,75
4	0,016	6436,3	6,65	5,84	2,34
5	0,008	6421,0	7,19	6,13	6,25
6	0,008	6428,5	6,81	5,98	4,56
7	0,008	6428,9	6,66	5,84	4,53
8	0,004	6421,1	7,26	5,98	6,49
9	0,004	6424,9	6,79	5,91	5,74
10	0,004	6421,1	7,26	5,98	6,49
11	0,002	6424,9	6,79	5,91	5,74
12	0,002	6423,0	6,99	5,94	6,19
13	0,002	6424,9	6,79	5,91	5,74
14	0,001	6422,9	6,99	5,94	6,19
15	0,001	6423,9	6,88	5,92	5,99
16	0,001	6424,9	6,79	5,91	5,74
17	0,0005	6423,9	6,88	5,92	5,99
18	0,0005	6424,4	6,83	5,916	5,87
19	0,0005	6423,95	6,88	5,925	5,99
20	0,00025	6424,4	6,83	5,916	5,87
21	0,00025	6424,2	6,85	5,920	5,931
22	0,00025	6424,2	6,85	5,916	5,925
23	0,000125	6424,1	6,85	5,914	5,94
24	0,000125	6424,24	6,84	5,911	5,91
25	0,000125	6424,12	6,85	5,914	5,94
26	0,0000625	6424,23	6,84	5,912	5,912
27	0,0000625	6424,18	6,85	5,913	5,926
28	0,0000625	6424,163	6,85	5,912	5,929
29	0,00003125	6424,159	6,85	5,911	5,929

первых пяти итерациях, затем оно уменьшалось после каждой трех итераций. Результаты приведены в табл. 2. После пятой итерации функция $\Phi[x(t)]$ имеет два локальных максимума; один — в точке $t=0,27 T$, второй — на конце интервала управления. Видно, что процесс и здесь принял колебательный характер, свидетельствующий о том, что шаг δu слишком велик.

Пятый расчет (табл. 3) отличался от четвертого тем, что среднее значение δu уменьшалось (после пятой итерации) в 1,2 раза после каждой итерации, однако это уменьшение прекращалось, когда δu достигла значения 0,000067. Это значение было выбрано по результатам четвертого расчета: при $\delta u=0,0000625$ процесс решения терминальной задачи стал относительно монотонным, значение $x^1(T)$ падает, а $\max \Phi \sim 5,9$. И в пятом расчете при $\delta u \simeq 0,000067$ началось монотонное падение $x^1(T)$ при сравнительно

Т а б л и ц а 3

ν	S	$x'(T)$	$\bar{F}(p)$	F'	F''
0	0,016	6443,68	7,35	6,44	—
1	0,016	6446,35	6,98	6,13	—
2	0,016	6448,29	6,65	5,84	—
3	0,016	6434,16	6,98	6,13	2,75
4	0,016	6436,27	6,65	5,84	2,34
5	0,013	6421,05	7,19	6,13	6,25
6	0,011	6433,66	6,71	5,89	3,01
7	0,0093	6423,04	7,02	6,09	5,98
8	0,0077	6432,85	6,74	5,92	3,53
9	0,0064	6429,69	6,63	5,81	4,25
10	0,0054	6423,39	6,91	5,92	6,04
11	0,0045	6428,62	6,65	5,83	4,59
12	0,0037	6424,24	6,83	5,91	5,85
13	0,0031	6420,76	7,24	5,98	6,43
14	0,0026	6423,64	6,88	5,91	5,99
15	0,0022	6426,14	6,72	5,87	5,35
16	0,0018	6424,05	6,84	5,91	5,89
17	0,0015	6422,36	7,03	6,94	6,23
18	0,0012	6423,77	6,87	5,92	5,96
19	0,0010	6424,97	6,77	5,90	5,67
20	0,00087	6423,99	6,85	5,91	5,91
21	0,00072	6423,14	6,94	5,93	6,09
22	0,00060	6423,82	6,86	5,92	5,95
23	0,0005	6424,40	6,81	5,91	5,81
24	0,00042	6423,92	6,85	5,91	5,92
25	0,00035	6423,79	6,86	5,91	5,95
26	0,00029	6424,12	6,83	5,90	5,87
27	0,00024	6423,85	6,85	5,91	5,93
28	0,00020	6423,91	6,85	5,90	5,92
29	0,00017	6423,72	6,86	5,91	5,95
30	0,00014	6423,88	6,85	5,91	5,92
31	0,00012	6423,77	6,86	5,91	5,94
32	0,000097	6423,88	6,85	5,91	5,92
33	0,000081	6423,79	6,85	5,91	5,94
34	0,000067	6423,81	6,85	5,906	5,930
35		6423,81	6,85	5,905	5,930
36		6423,81	6,85	5,904	5,929
37		6423,80	6,85	5,903	5,929
38		6423,80	6,85	5,902	5,928
41		6423,78	6,85	5,900	5,929
44		6423,77	6,84	5,897	5,929
47		6423,75	6,84	5,894	5,929
50		6423,74	6,84	5,890	5,929
51		6423,73	6,84	5,890	5,929

Т а б л и ц а 4

ν	$x^1(T)$ предск.	$x^1(T)$ фактич.	F'	F''	$ \delta u(t) $ среднее	K
0		6443,68	6,440	—	0,016	1
1	6430,72	6430,18	6,793	—	0,016	1
2	6421,27	6421,51	6,753	5,193	0,016	1
3	6413,62	6414,01	6,627	5,290	0,016	1
4	6407,38	6407,80	6,407	5,500	0,016	1
5	6896,82	6397,37	5,958	6,376	0,013	1
6	6395,93	6396,50	6,645	5,962	0,011	1
7	6396,00	6396,32	5,989	6,460	0,013	1
8	6396,07	6396,52	6,652	5,652	0,0076	2
9	6396,00	6396,29	6,008	5,971	0,0074	2
10	6396,01	6396,06	5,907	5,991	0,0074	3
11	6395,96	6396,02	5,901	5,935	0,0096	4

стабильном значении $\max \Phi$. Однако продолжать вычисления подобным образом практически бессмысленно. Даже если предположить, что характер процесса сохранится, то для достижения значения $x^1(T) = 6396,00$ понадобится ~ 4000 итераций. А ведь решение задачи только после этого, собственно, и начнется, причем в гораздо более сложных условиях: $\Phi[x(t)]$ будет уже не «двугорбой» функцией, а функцией с «плато».

Для сравнения в табл. 4 подробно показан начальный этап решения той же задачи методом проекции градиента. Приведены следующие величины: ν — номер итерации, предсказанное значение $x^1(T)$, F' и F'' — значения $\Phi[x(t)]$ в двух точках локальных максимумов, среднее значение $|\delta u(t)|$ и K — число точек аппроксимации в формуле (35). Стоит отметить, что механизм постепенного увеличения K сработал с небольшим опозданием: уже на 6-й итерации $\Phi[x(t)]$ стала «двугорбой», и следовало увеличить K с 1 до 2. Таким образом, терминальная задача была довольно легко решена, и уже с 9-й итерации начался процесс минимизации.

§ 40. Некорректные задачи оптимального управления.

Регуляризация численного решения

В работах [79], [80] было обращено внимание на то, что задача оптимального управления некорректна, т. е. ее решение может быть изменено на конечную величину без изменения значений тех функционалов, в терминах которых поставлена задача. Правда, в общем случае это есть возмущение управления на множестве меры нуль, что особенно серьезного значения не имеет. Однако существует класс задач (и он не так уж узок), в которых некорректность связана с возможностью при сколь угодно малом изменении значений

функционалов изменить управление на конечную величину на множестве положительной меры.

В качестве характерного примера подобной задачи рассмотрим задачу о вертикальном подъеме ракеты (см. §§ 28, 29). Ищется функция $u(t)$, минимизирующая значение функционала $F_0[u(\cdot)] \equiv 1 - x^1(T)$ при условиях $F_1[u(\cdot)] \equiv x^2(T) - 1,49 = 0$ и $0 \leq u(t) \leq U$. Здесь $x(t) = \{x^1, x^2, x^3\}$ — решение задачи Коши $\dot{x} = f(x, u)$:

$$\begin{aligned} \dot{x}^1 &= -u; & \dot{x}^2 &= x^3; & \dot{x}^3 &= -g + [Vu - Qe^{-\alpha x^1}(x^3)^2]/x^1, \\ x^1(0) &= 1; & x^2(0) &= 0; & x^3(0) &= 0; & 0 \leq t \leq T \end{aligned} \quad (1)$$

(g, V, α, Q, U, T — заданные числа, см. § 28). Решение этой задачи имеет следующий вид:

$$u(t) = \begin{cases} U & \text{при } 0 \leq t \leq t_1, \\ u^*(t) & \text{при } t_1 < t < t_2, \\ 0 & \text{при } t > t_2, \end{cases}$$

где $u^*(t)$ ($0 < u^*(t) < U$) — некоторая функция, вычисление которой описано в § 28. Интервал (t_1, t_2) носит в приложениях название «участка особого решения». Тогда существует число $a > 0$ такое, что

$$0 \leq \tilde{u}(t) \equiv u^*(t) + a \sin kt \leq U \quad \text{при } t \in (t_1, t_2).$$

В то же время нетрудно показать, что

$$F_i[\tilde{u}(\cdot)] = F_i[u^*(\cdot)] + O(1/k), \quad i = 0, 1, \quad (2)$$

т. е. при достаточно большой частоте возмущения k функционалы (и вообще траектория $x(t)$) изменяются сколь угодно мало. Это и есть некорректность задачи.

Проверить свойство (2) несложно: введем на $[0, T]$ сетку точек $t_n = n\Delta t$, $n = 0, 1, \dots, N$, $\Delta t = T/N$, и обозначим $x_n = x(t_n)$. Тогда x_n удовлетворяют уравнениям

$$x_{n+1} = x_n + \int_{t_n}^{t_{n+1}} f[x(t), u(t)] dt = x_n + \int_{t_n}^{t_{n+1}} f[x_n, u(t)] dt + O(\Delta t^2). \quad (3)$$

В силу линейности $f(x, u)$ по u (в задаче (1))

$$\int_{t_n}^{t_{n+1}} f[x_n, u(t) + a \sin kt] dt = \int_{t_n}^{t_{n+1}} f[x_n, u(t)] dt + O(\Delta t/k),$$

и величину $O(\Delta t/k)$ можно, выбрав достаточно большое k , считать величиной $O(\Delta t^2)$. Однако решение системы разностных уравнений

$$y_{n+1} = y_n + \int_{t_n}^{t_{n+1}} f[y_n, u(t)] dt + O(\Delta t^2), \quad y_0 = x_0, \quad n = 0, 1, \dots, N \quad (4)$$

расходится с решением системы (3) на величину $O(\Delta t)$. В силу произвольности Δt сделанное выше утверждение о некорректности задачи (1) доказано. Разумеется, для аккуратности следует еще изменить управление (на $O(\Delta t)$) так, чтобы условие $F_1[\bar{u}(t)] = 0$ было выполнено. Это, конечно, возможно (за счет дополнительного изменения F_0 на $O(\Delta t)$).

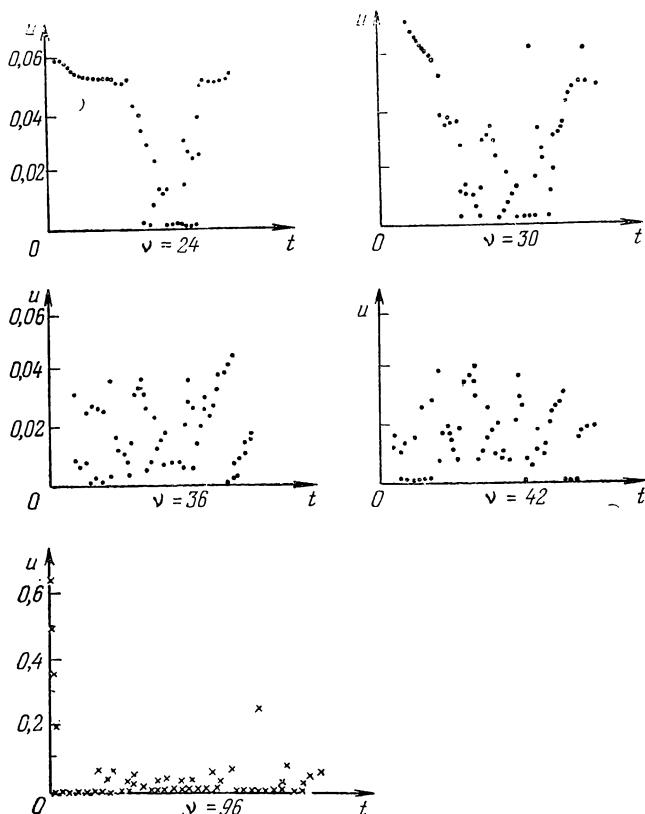


Рис. 58.

Этот факт — возможность значительных вариаций управления, приводящих к очень малым изменениям функционалов, — имеет очень важные с вычислительной точки зрения последствия. Именно с ним связано то, что часто удается получить решение, очень точное по значениям функционалов, в то время как сама управляющая функция оказывается довольно грубым приближением к точной. Читатель без труда заметит последствия этого свойства задач оптимального управления в примерах найденных приближенных решений задач. Здесь же мы рассмотрим результаты вычислительного эксперимента, поставленного специально с целью получить

проявление некорректности возможно более ярко и в этих условиях испытать разработанный автором метод борьбы с этой неприятностью (регуляризация). Специальные условия эксперимента состояли лишь в том, что задача (1) решалась (методом последовательной линейаризации) на сетке с очень мелким шагом, содержащей ~ 500 счетных интервалов; при этом на участок $[0, t_2]$ приходилось ~ 200 точек. Хорошо известно, что некорректность задачи проявляется тем сильнее, чем выше размерность конечномерного пространства, аппроксимирующего функциональное (чем меньше шаг сетки, тем больше «частота» k , которую можно реализовать в пространстве кусочно постоянных сеточных функций). На рис. 58 показана эволюция функции $u^\nu(t)$ на некотором участке $(t', t'') \in [t_1, t_2]$ (ν — номер итерации) в процессе поиска решения. Видно, как постепенно график функции $u(t)$ портится и в конце концов (при $\nu=96$) превращается в набор « δ -функций». В то же время эволюция функционалов F_0, F_1 вполне разумна, и их значения при $\nu \approx 90$ почти не отличаются от значений в точном решении (см. табл. 1, задача 1). Однако, если иметь в виду содержательную интерпретацию полученных в расчетах результатов, то представленное на рис. 58 практически оптимальное управление $u(t)$ совершенно неудовлетворительно. Надо сказать, что найденные численно оптимальные управления очень редко используются на практике в том виде, в каком они получены математиком (даже если они и точные). Обычно задачи оптимального управле-

Т а б л и ц а 1

ν	Задача 1		Задача 2	Задача 3	
	F_0	$x^2(T)$	F_0	F_0	$x^2(T)$
0	0,70000		0,70000	0,70000	
6	0,68869	1,49150	0,68501	0,68020	1,4872
12	0,68794	1,48998	0,68313	0,68010	1,48930
18	0,68653	1,48970	0,68120	0,68000	1,48907
24	0,68425	1,48993	0,68004	0,68001	1,48973
30	0,68166	1,48941	0,67950	0,68000	1,48985
36	0,68051	1,48989	0,67920		
42	0,67975	1,48895	0,67903		
48	0,67936	1,48943	0,67892		
54	0,67915	1,48988	0,67886		
60	0,67896	1,48942	0,67881		
66	0,67887	1,48981	0,67882		
72	0,67879	1,48963			
78	0,67873	1,48955			
84	0,67869	1,48940			
90	0,67865	1,48951			
96	0,67868	1,48975			

ния ставятся для некоторых математических моделей реального объекта, и их решения служат лишь ориентиром, они должны быть еще аппроксимированы теми средствами, которыми реально располагает конструктор. Например, точное решение в задаче (1) на участке $[t_1, t_2]$ не может быть описано сравнительно простым аналитическим выражением, и создание автоматики, обеспечивающей расход горючего в точном соответствии с формой оптимального $u(t)$, было бы чрезвычайно трудным. К счастью, это не нужно. Визуальный анализ точного решения наводит на мысль, что среди существенно более простых конструкций управления типа

$$u(t) = \begin{cases} U & \text{при } 0 \leq t < t_1, \\ a & \text{при } t_1 \leq t < t_2, \\ 0 & \text{при } t_2 \leq t \leq T \end{cases} \quad (5)$$

(здесь t_1, t_2, a — некоторые параметры) найдется функция, мало уступающая по значению F_0 оптимальной. Для t_1, t_2, a можно (зная точное решение) легко указать приближенные значения, которые затем без труда уточняются решением задачи (1) в классе управлений (5). Еще более точная аппроксимация будет достигнута, если на (t_1, t_2) взять двухпараметрическое семейство функций $a + b(t - t_1)$ или разбить интервал (t_1, t_2) на (t_1, t') и (t', t_2) и на каждом из них взять в качестве a разные постоянные a', a'' и т. д. Стоит иметь в виду, что сам процесс поиска оптимального управления содержит информацию о том, какого сорта вариации управления слабо влияют на значения функционалов. В частности, в примере (1), как мы увидим, четкое выделение разрывов в $u(t)$ (в точках t_1, t_2) не обязательно: разрывы можно «размазать». Решение на рис. 58 плохо еще и потому, что, глядя на него, едва ли можно догадаться, что в простом классе функций (5) есть очень точные аппроксимации оптимального управления. Естественно, возникает желание «регуляризовать» численное решение, т. е. включить в постановку задачи еще и требование, чтобы функция $u(t)$ была достаточно простой. Обычно трудно бывает придать этому качественному требованию однозначную количественную формулировку — в этом одна из сложностей процедуры регуляризации. Однако математический аппарат здесь уже разработан (см., например, [79], [80], [81]). Стандартный прием состоит в добавлении к минимизируемому функционалу $F_0[u(\cdot)]$ малой регуляризующей добавки: решается задача $\min F_0[u(\cdot), \alpha]$, где

$$F_0[u(\cdot), \alpha] \equiv F_0[u(\cdot)] + \alpha \int_0^T |\dot{u}|^2 dt, \quad (6)$$

причем $\alpha > 0$ — малое число. Такая замена приводит к тому, что среди мощного множества управлений, почти не отличаю-

щихся друг от друга по значениям F_0, F_1 , предпочтение отдается более гладкому, имеющему меньшее значение $\int \dot{u}^2 dt$. Автором была предпринята попытка использовать этот подход, однако она оказалась неудачной: трудно подобрать нужное значение α . При слишком малых α не получалось гладкого решения, при больших α — значение F_0 заметно превосходило минимальное. Требуется производить подбор α , что связано с увеличением трудоемкости расчета. Поэтому была предложена другая реализация общей идеи регуляризации. Построенный на ее основе алгоритм позволил получить хорошее численное решение без увеличения затрат машинного времени.

Сначала попытаемся понять, как возникает та картина, которая изображена на рис. 58. Напомним, что процесс решения задачи состоит в последовательном варьировании управления: $u(t)$ переходит в $u(t) + \delta u(t)$, где $\delta u(t)$ — решение задачи

$$\min_{\delta u(\cdot)} \int_0^T w_0(t) \delta u(t) dt, \quad (7)$$

при условиях

$$F_1[u(\cdot)] + \int_0^T w_1(t) \delta u(t) dt = 0, \quad (8)$$

$$s^-(t) \leq s(t) \leq s^+(t)$$

($w_0, w_1(t)$ — производные Фреше функционалов F_0, F_1). Решение задачи (7), (8) определяется некоторым вектором $\{1; g\}$ и имеет вид

$$\delta u(t) = \begin{cases} s^-(t), & \text{если } w_0(t) + gw(t) > 0, \\ s^+(t), & \text{если } w_0(t) + gw_1(t) < 0. \end{cases} \quad (9)$$

В данной задаче решение имеет нечто вроде численного аналога δ -функции: величина U очень велика по сравнению с $u(t)$ на (t_1, t_2) . В процессе итераций управление на интервале (t_1, t_2) быстро становится, так сказать, оптимальным: в том смысле, что существует число g , для которого

$$w_0(t) + gw_1(t) \approx 0 \quad \text{при } t \in (t_1, t_2). \quad (10)$$

Подобное соотношение выполняется уже после нескольких первых итераций, однако в целом управление еще не оптимально. Знак $w_0(t) + gw_1(t)$ становится на (t_1, t_2) , по существу, случайной величиной, зависящей, в частности, и от погрешностей конечно-разностной аппроксимации дифференциальных уравнений. Следовательно, и $\delta u(t)$ на (t_1, t_2) становится, в известной мере, случайной. В сочетании с некорректностью задачи эта случайность и приводит

к тому, что изображено на рис. 58. Приняв такое объяснение происхождения численной некорректности, построим алгоритм регуляризации, сосредоточив внимание на тех участках интервала времени, где $w_0(t) + gw_1(t) \approx 0$.

А л г о р и т м р е г у л я р и з а ц и и. Пусть задача (7), (8) решена, найден соответствующий вектор $\{1, g\}$, и управление проварьировано: $u(t) := u(t) + \delta u(t)$. Выделим на $[0, T]$ множество M :

$$t \in M: |w_0(t) + gw_1(t)| \leq \varepsilon \sqrt{w_0^2(t) + w_1^2(t)}. \quad (11)$$

Теперь найдем дополнительную вариацию $\delta u(t)$ решением задачи

$$\min_{\delta u(\cdot)} \int_0^T [\ddot{u}(t) + \delta \ddot{u}(t)]^2 dt \quad (12)$$

при условиях

$$\begin{aligned} \int_0^T w_1(t) \delta u(t) dt &= 0, \\ s^-(t) &\leq \delta u(t) \leq s^+(t), \\ \delta u(t) &\equiv 0 \quad \text{при } t \notin M, \end{aligned} \quad (13)$$

т. е. $s^-(t) = s^+(t) = 0$ при $t \notin M$.

Таким образом, регуляризирующая вариация $\delta u(t)$ производится только там, где в линейном приближении вариация управления с точки зрения функционалов F_0, F_1 может быть совершенно произвольной; в самом деле, если $w_0(t) + gw_1(t) = 0$, то из

$$\int w_1 \delta u dt = 0 \quad \text{следует} \quad \int w_0 \delta u dt = 0.$$

Разумеется, в расчетах $\varepsilon > 0$, его назначение и регулирование обсуждаются ниже. Заметим, что в задаче (12), (13) границы $s^-, s^+(t)$ не совпадают с $s^-, s^+(t)$ в (8). При их вычислении, кроме обычных соображений о малости δu и выполнении условия $0 \leq u + \delta u \leq U$, используется еще одно. Задача (12), (13) есть задача квадратичного программирования, однако в расчетах она решалась итерационным методом, причем на каждой итерации решалась задача линейного программирования (являющаяся ли-неаризацией (12)):

$$\min_{\delta u(\cdot)} \int_0^T \{-\ddot{u}(t)\} \delta u(t) dt, \quad (12^*)$$

при условиях (13). В классе сеточных функций $u_{n+1/2}$, $\ddot{u}_{n+1/2} = \{u_{n+1/2} - 2u_{n+1/2} + u_{n-1/2}\}$, и на величину $\delta u_{n+1/2}$ накладывалось еще ограничение

$$|\delta u_{n+1/2}| \leq \frac{1}{4} |\ddot{u}_{n+1/2}|.$$

Это условие есть аналог известного условия устойчивости явной схемы для уравнения теплопроводности. Мы же, по существу, используем процесс выглаживания, аналогичный счету по явной схеме. После того как задача (12*) (13) решена, найдено новое управление $u(t) + \delta u(t)$, пересчитываются $\ddot{u}_{n+1/2}$, s^- , $s^+(t)$, снова решается задача (12*), (13), и так заданное число раз K . Одновре-

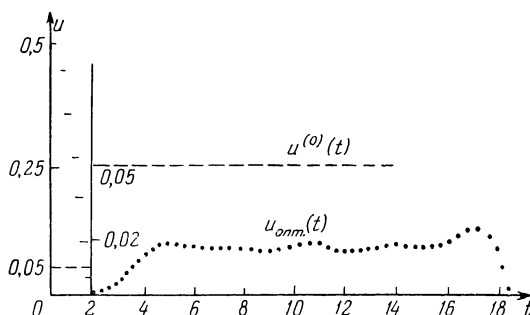


Рис. 59.

менно регулируется и число ϵ . Обозначим последовательно определяемые решением (12*) (13) вариации через $\delta u^{(k)}$, $k=1, 2, \dots, K$. Когда была найдена основная вариация $\delta u(t)$ решением задачи (7), (8), была вычислена и величина выигрыша $\delta F_0 [\delta u(\cdot)] = \int_0^T w_0 \delta u dt$. Вариации $\delta u^{(k)}(t)$, имеющие целью выглаживание управления, приводят, конечно, к некоторым вариациям $\delta F_0^{(k)} = \int_0^T w_0 \delta u^{(k)} dt$. Зададимся теперь некоторой величиной (например, как было в расчетах, $0,05 |\delta F_0|$), которую мы «жертвуем» на гладкость. Если после очередной k -й итерации $\left| \sum_{j=1}^k \delta F_0^{(j)} \right| \leq 0,05 |\delta F_0|$, то число ϵ увеличивается, т. е. множество M расширяется. Это характерный прием: нам нужно выбрать число ϵ так, чтобы выглаживание u на M не очень сильно влияло на основной процесс минимизации F_0 и с этой точки зрения выбор 5% от δF_0 едва ли требует обоснования. Прямую связь между ϵ и желаемыми 5% δF_0 указать невозможно, поэтому подбор ϵ осуществляется простым алго-

ритмом, основанным на использовании обратной связи с тем, что получается в расчетах. Разумеется, 5% — это достаточно условная мера, и особая точность здесь не нужна. В табл. 1 показана эволюция F_0 в процессе решения той же задачи I с применением регуляризации (задача II). Значения F_1 практически не отличаются от значений F_1 в задаче I. На рис. 59 показана найденная в этом расчете функция $u(t)$: она уже годится для содержательного истолкования. На рис. 60 показана (на примере задачи того же типа, что I, но с другим значением параметра U) роль числа регуляризирующих итераций K (в одном расчете 5, в другом 10). По значениям функционалов F_0 , F_1 эти два расчета почти совпадают,

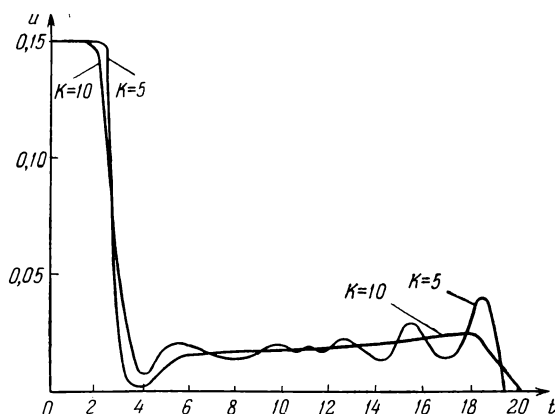


Рис. 60.

поэтому в табл. 1 приведены данные только о решении одной из этих задач (задача III).

Теперь обсудим возможные причины неудачи расчетов с функционалом $F_0[u(\cdot), \alpha]$ (6). Видимо, дело в том, что искомое решение $u(t)$ — разрывно: ведь на разрывной функции $\int u^2 dt = \infty$. Если бы точки разрыва t_1, t_2 были заранее известны, можно было бы использовать функционал

$$F_0[u(\cdot)] + \alpha \int_0^{t_1-\varepsilon} \dot{u}^2 dt + \alpha \int_{t_1+\varepsilon}^{t_2-\varepsilon} \dot{u}^2 dt + \alpha \int_{t_2+\varepsilon}^T \dot{u}^2 dt. \quad (6^*)$$

К сожалению, обычно положение разрывов заранее не известно. Существует прием, который, как кажется, позволяет фиксировать точки разрыва. Именно, от системы $\dot{x} = f(x, u)$, введя дополнитель-

ную компоненту управления v и новое время τ , перейдем к системе

$$\begin{aligned}\frac{dx}{d\tau} &= v(\tau) f(x, u), & x(0) &= x_0, & 0 \leq \tau \leq 1, \\ \frac{dt}{d\tau} &= v(\tau), & t(0) &= 0; & v(\tau) \geq \alpha > 0, \\ & & t(1) &= T\end{aligned}\quad (14)$$

(появляется и дополнительное условие $t(1)=T$). В новом времени τ положение разрывов можно задать более или менее произвольно: например, $\tau_1=1/3$; $\tau_2=2/3$. Однако этот формальный путь чреват неприятностями. Дело в том, что после перехода к системе (14) задача приобрела тривиальную неединственность: можно выбрать континуум разных функций $v(\tau)$ и к каждой из них подобрать $u(\tau)$ так, что эти, внешне разные, пары функций $\{v(\tau), u(\tau)\}$ после перехода к физическому времени t дадут одну и ту же функцию $u(t)$, одни и те же F_0, F_1 . Следствием этого является случайная, в сущности, эволюция $v(\tau)$ в процессе решения задачи. Автору неизвестны никакие соображения в пользу того, что эволюция $v(\tau)$ будет происходить так, чтобы формальные точки разрыва τ_1, τ_2 совпали с действительными, т. е. чтобы было

$$\int_0^{\tau_1} v(\tau) d\tau = t_1, \quad \int_0^{\tau_2} v(\tau) d\tau = t_2. \quad (15)$$

Не видно и способов, которыми можно было бы добиваться выполнения (15). Попытки использовать замену (14) действительно привели к бессмысленной эволюции $v(\tau)$ в процессе поиска минимума. Впервые регуляризация численного решения задачи оптимального управления была осуществлена в работе [81], причем использовался именно функционал $F_0[u(\cdot), \alpha]$ (6). Решалась, кстати, задача, совпадающая с (1), однако в более простой ситуации: решение искалось только на интервале (t_1, t_2) (оптимальные значения t_1 и t_2 можно искать подбором, причем каждый акт подбора связан с решением вариационной задачи на (t_1, t_2)). Поэтому ограничение $0 \leq u \leq U$ отсутствовало, и применялся метод проекции градиента (классический вариант, § 18). В этом случае вариация $\delta u(t)$ имеет вид

$$\delta u(t) = -w_0(t) + \lambda w_1(t), \quad (16)$$

т. е. является гладкой малой функцией. Итераций нужно не так уж много, и сумма небольшого числа функций типа (16) не может быть очень негладкой. Поэтому некорректность в расчетах проявилась слабо (см. [81], фиг. 1), и с ней легко было справиться переходом к (6), причем величина α может выбираться с большой степенью неопределенности. Кстати, с этим же связано и то, что не-

корректность задач оптимального управления была замечена чисто теоретически в работах [79], [80], хотя численное решение таких задач началось много раньше. Дело в том, что большей частью решались относительно простые задачи классического типа, и применялся в основном метод проекции градиента, в котором вариация $\delta u(t)$ находилась по формулам, аналогичным (16). Напомним, что $w(t) = f^*[x(t), u(t)]\phi(t)$, где и $x(t)$, и $\phi(t)$ — решения дифференциальных уравнений, т. е. гладкие функции. Другой причиной является использование сравнительно грубых сеток, позволяющих получить хорошую аппроксимацию оптимального управления, но не приводящих еще к сильному проявлению некорректности, хотя некоторые ее слабые следы заметны во многих приближенных решениях. Заметим, что во всех остальных задачах, решение которых приведено в этой книге, регуляризация не применялась. Понятие некорректности задачи первоначально возникло в связи с так называемыми обратными задачами. Типичными обратными задачами являются следующие:

1) зная температуру тела в данный момент времени $t=0$, определить распределение температуры в прошлом, в момент $t_1 < 0$ (обратная задача теплопроводности);

2) зная величину гравитационного поля на поверхности Земли, определить распределение массы внутри Земли (обратная задача гравиметрии).

Характерная трудность решения таких задач состоит в том, что незначительная неопределенность в исходных данных связана с очень большой неопределенностью в решении, и неопределенность эта нетерпима, так как решение представляет собой информацию о реальной действительности. Регуляризация в подобных задачах состоит, грубо говоря, в том, что к условию задачи добавляется еще некоторая качественная информация о решении, например, что решение — достаточно гладкая функция.

В задачах оптимального управления ситуация несколько иная. Здесь решение не есть информация о мире, оно является лишь рекомендацией о наиболее эффективном поведении (управлении). Поэтому неопределенность ответа не очень страшна. Более того, если обнаруживается, что существует мощное семейство управлений, приводящих к одним и тем же практически оптимальным результатам, то это следует расценивать как благоприятное обстоятельство: ведь этим облегчается задача аппроксимации оптимального управления фактически реализуемыми средствами. Наконец, отметим связь используемого в наших расчетах метода регуляризации с одной идеей решения экономических задач. Часто в экономике возникают задачи, в которых нужно минимизировать не один показатель (функционал) $F_0(u)$, а несколько: $F_{0,1}(u)$, $F_{0,2}(u), \dots$, причем они занумерованы (вторым индексом) в порядке важности. Достаточно осмысленный подход к подобным зада-

чам состоит в следующем. Сначала решается задача на $\min F_{0,1}(u)$, остальные функционалы $F_{0,2}(u), \dots$ пока игнорируются. Пусть $\Lambda_1 = \min F_{0,1}(u)$. Далее выбирается некоторая величина $\epsilon_1 > 0$ — часть $F_{0,1}$, которая «жертвуется» на достижение второстепенных целей, и решается задача

$$\min_u F_{0,2}(u) \quad \text{при условии} \quad F_{0,1} \leq \Lambda_1 + \epsilon_1.$$

Затем таким же образом часть $F_{0,2}$ «жертвуется» на минимизацию $F_{0,3}$ и т. д.

§ 41. Решение обратных задач математической физики. Вариационный подход

В последнее время интенсивно развивается теория так называемых некорректных задач. Мы не будем приводить общих определений, а рассмотрим достаточно характерный пример — обратную задачу теплопроводности. Итак, пусть функция $v(t, x)$ в области $0 \leq x \leq 1$, $0 \leq t \leq T$ является решением краевой задачи с начальными данными при $t=0$:

$$\frac{\partial v}{\partial t} = \frac{\partial^2 v}{\partial x^2}, \quad v(0, x) = u(x), \quad v(t, 0) = 0; \quad v(t, 1) = 0. \quad (1)$$

Обозначим $w(x) = v(T, x)$. Тогда можно говорить, что задача (1) определяет линейное отображение функции $u(x)$ в функцию $w(x)$. Обозначим его $w = R(T)u$. *Прямая задача*, состоящая в вычислении w по известному u , хорошо изучена, и легко решается, например, классическим методом Фурье. В более сложных задачах подобного рода можно воспользоваться известными разностными методами. Задача (1) устойчива: существует постоянная C (в данном случае $C < 1$) такая, что если $w^* = Ru^*$ и $\|u^* - u\| \leq \epsilon$, то $\|w^* - w\| \leq C\epsilon$. (Другими словами, $\|R\| \leq C$.) Однако современная техника и естествознание потребовали решения обратных задач:

П о с т а н о в к а о б р а т н о й з а д а ч и. Пусть известна функция $w(x)$, причем неточно, а с некоторой ошибкой δ . Другими словами, известна некоторая функция $w(x)$, но об «истинной» функции $\tilde{w}(x)$ известно лишь, что $|w(t) - \tilde{w}(t)| \leq \delta$. Определить функцию $u(x)$ так, что

$$\|Ru - w\| \leq \delta. \quad (2)$$

Уточним суть дела. Мы имеем некоторое множество W в пространстве функций w — оно состоит из всех функций $w^*(x)$, удовлетворяющих неравенству $|w(t) - w^*(t)| \leq \delta$. Тогда можно говорить о его прообразе U — множестве функций $u(x)$, удовлет-

воряющих (2), и в качестве решения можно взять любой из элементов $u \in U$. Но так поставленная задача бессмысленна: среди функций $u \in U$ есть функции, отличающиеся друг от друга на произвольно большую величину. Иначе говоря, поперечник (функциональный) множества U бесконечен при сколь угодно малой величине δ . Этот факт, следующий из хорошо известного примера Адамара, обычно служил доводом, показывающим бессмысленность и неразрешимость обратной задачи. Однако подобные задачи были поставлены, например, в связи с попытками по возмущениям гравитационного поля на поверхности Земли получить сведения о каких-то деталях ее глубинного строения (обратная задача теории потенциала). Решение их оказалось возможным (во всяком случае, принципиально) благодаря уточнению постановки задачи: кроме (2), от функции u требуется еще выполнение некоторых качественных условий, носящих не очень четкую и однозначную форму, например, чтобы $u(x)$ была достаточно гладкой функцией, или что-нибудь в этом же роде. Была разработана и математическая теория учета подобных дополнительных требований к $u(x)$ — хорошо известная сейчас *теория регуляризации некорректных задач* (А. Н. Тихонов, В. К. Иванов, М. М. Лаврентьев и др.). На основе этой теории разрабатывались численные методы решения обратных задач. Приведем характерные конструкции.

1. Найти функцию $u(x)$ решением задачи

$$\min_{u(\cdot)} \left\{ \|Ru - w\|^2 + \alpha \left\| \frac{du}{dx} \right\|^2 \right\}.$$

2. Найти функцию $u(x)$, реализующую

$$\min_{u(\cdot)} \|Ru - w\| \quad \text{при} \quad \left\| \frac{du}{dx} \right\| \leq A.$$

3. Найти числа c_1, c_2, \dots, c_N из условия

$$\min_c \left\| R \sum_{k=1}^N c_k \sin k\pi x - w(x) \right\|.$$

4. Найти $u(x)$ решением задачи

$$\min_{u(\cdot)} \left\| \frac{du}{dx} \right\| \quad \text{при} \quad \|Ru - w\| \leq \delta.$$

Здесь α, A, N — параметры регуляризации, выбор которых далеко не прост и существенно влияет на результат. Все перечисленные выше задачи являются, в сущности, вариационными задачами. Наконец, отметим и развиваемый французским математиком Лионсом *метод квазиобращения*. Применительно к задаче теплопроводности он состоит в следующем: функция $u(x)$ определяется

как решение регуляризованного обратного уравнения теплопроводности

$$\frac{\partial v}{\partial t} = \frac{\partial^2 v}{\partial x^2} + \varepsilon \frac{\partial^4 v}{\partial x^4}; \quad v(T, x) = w(x); \quad v(t, 0) = v(t, 1) = 0. \quad (3)$$

Решив эту задачу, полагаем $u(x) = v(0, x)$ (см. [45]). Обратная задача теплопроводности является удобным методическим примером некорректной задачи. Многие вычислители (например, Лионс) используют ее для отработки приближенных методов. Здесь мы намерены также использовать эту задачу в качестве теста, поэтому стоит разобрать ее несколько подробнее и выявить характерные трудности.

Прежде всего напомним явное выражение для решения прямой задачи. Разложим $v(0, x)$ в ряд Фурье:

$$v(0, x) = \sum_{k=1}^{\infty} c_k \sin k\pi x.$$

Тогда $v(t, x) = \sum_{k=1}^{\infty} c_k e^{-\lambda_k t} \sin k\pi x$, где $\lambda_k = k^2 \pi^2$, а для оператора $R(T)$ получим явное выражение

$$R(T) \sum_{k=1}^{\infty} c_k \sin k\pi x = \sum_{k=1}^{\infty} e^{-\lambda_k T} c_k \sin k\pi x = w(x). \quad (4)$$

Из него видна основная причина трудности решения некорректных задач: информация о функции $u(x)$, содержащаяся в решении $v(t, x)$ задачи (4), быстро исчезает и теряется на фоне ошибок

Таблица 1

$e^{-\lambda_k T}$					
$T \backslash k$	1	2	3	4	5
0,01	0,9	0,67	0,41	0,2	0,08
0,02	0,81	0,45	0,23	0,04	0,0064
0,1	0,37	0,02	10^{-4}	10^{-7}	10^{-11}
$e^{-\lambda_k T}$					
$T \backslash k$	6	7	8	9	10
0,01	0,03	$7 \cdot 10^{-3}$	$1,6 \cdot 10^{-3}$	$3 \cdot 10^{-4}$	$5 \cdot 10^{-5}$
0,02	$9 \cdot 10^{-4}$	$5 \cdot 10^{-5}$	$3 \cdot 10^{-6}$	10^{-7}	$2 \cdot 10^{-9}$
0,1	10^{-16}				

измерения $w(x)$. Поэтому и восстановить ее трудно. Чтобы более наглядно представить себе темп «затухания информации», приведем таблицу значений величины $e^{-\lambda k T}$ для разных k и T .

Пусть мы каким-то способом нашли функцию $u(x)$, удовлетворяющую условию $Ru = w$. Но тогда и функция $u(x) + \delta e^{\lambda k T} \sin k\pi x$ тоже может претендовать на роль решения задачи, так как

$$R[u(x) + \delta e^{\lambda k T} \sin k\pi x] = Ru + \delta \sin k\pi x = w + O(\delta),$$

и при достаточно большом k неоднозначность определения u становится сколь угодно большой. Все методы решения некорректных задач состоят в том, чтобы так или иначе запретить появление в искомом ответе высоких гармоник с большими и даже просто конечными коэффициентами. Но что такое «высокая частота», начиная с какого номера k нужно функцию $\sin k\pi x$ считать лишней, только портящей решение? Это, конечно, зависит от T . При $T=0,1$, и при δ , допустим, порядка 10^{-3} , и при условии, что искомое $v(0, x) \sim 1$, придется постараться обойтись без третьей гармоники: ведь добавление к $v(0, x)$ функции $\sin 3\pi x$ исказит $w(T, x)$ на величину $\sim \delta = 10^{-3}$, четвертую же гармонику можно добавить с коэффициентом 10^3 . Обратная задача теплопроводности с $T=0,1$ является основным тестом, используемым в известной монографии [45] для иллюстрации возможностей метода квазиобращения. Нам, однако, этот вариант задачи кажется методически не очень удачным: в такой задаче информация о $v(0, x)$ в $w(x)$ (учитывая ошибки $\sim \delta$), в сущности почти отсутствует. Это, кстати, подтверждают и приведенные в [45] результаты вычислений: решая обратную задачу на сетке с шагом $\Delta x = 1/50$, в [45] получают в качестве $u(x)$ (при достаточно большой разнице $\|Ru - w\|$, достигающей $\sim 5\%$ от w в лучших случаях) функцию типа $u(x) \simeq 10^8 \sin 6\pi x$ (при $|w(x)| \leq 1$ *).

Трудно представить себе задачу естествознания или технологии, в которой подобные результаты допускают содержательное истолкование. Рассматривать такие задачи как чисто методические, имеющие целью отработать соответствующие вычислительные методы, тоже нужно с большой осторожностью. Ведь эти задачи должны отражать существенные черты реальных, иначе велика вероятность сосредоточить усилия на преодолении трудностей, характерных именно для данного экстремального случая и не встречающихся в реальных ситуациях, и, наоборот, оставить без внимания последние. Поэтому мы будем экспериментировать при $T=0,01$, когда все-таки можно рассчитывать на более содержательные результаты. Вообще, разрабатывая методы решения некорректных задач, нужно достаточно четко представлять себе,

*) Кстати, такое решение обладает своеобразной неустойчивостью: искажив его функцией $10^2 \sin \pi x$ (т. е. введя в $u(x)$ погрешность $\delta u(x)$, $|\delta u| \simeq 10^{-6}|u|$), величину $\|R(u + \delta u) - w\|$ увеличим в $\sim 10^8$ раз.

какую, хотя бы качественно, ситуацию мы имеем в виду, так как от этого будет зависеть и арсенал привлекаемых вычислительных средств. Так, если все-таки считать задачу с $T=0,1$ характерным примером и пытаться достаточно аккуратно восстановить $u(x)$, придется предполагать $w(x)$ заданной очень точно, и чрезвычайно остро встает вопрос об ошибке численного интегрирования, об ошибках округления при вычислениях на ЭВМ и конечной разрядности чисел; их влиянием уже нельзя пренебречь. Можно, конечно, и в этой задаче ослабить остроту этих проблем, не претендуя на какую-то точность при восстановлении $u(x)$. Такой, в сущности, точки зрения придерживаются авторы [45]: судя по приведенным там графикам 1÷8 найденных $u(x)$, к этим функциям предъявляются следующие требования: они должны удовлетворять условию (2) (с не очень малым δ), и их можно благополучно (без аварийного останова) вычислить на ЭВМ.

Все перечисленные выше методы решения некорректных задач можно формально записать в виде

$$u = R_{\varepsilon}^{-1}w,$$

где R_{ε}^{-1} — регуляризованный обратный оператор R^{-1} . При этом R^{-1} неограничен, а R_{ε}^{-1} ограничен, и, что очень важно, R_{ε}^{-1} с хорошей точностью аппроксимирует R^{-1} на подпространстве гладких функций, существенно отличаясь от него на негладких. Точность аппроксимации и условная граница, отделяющая «гладкие» функции от «негладких», зависят, разумеется, от конкретного способа регуляризации и параметра ε . Однако, кроме самого факта ограниченности R_{ε}^{-1} , очень важной характеристикой является его норма $\|R_{\varepsilon}^{-1}\|$. От ее величины зависит соотношение между точностью задания $w(x)$ и точностью ответа $v(0, x) = R_{\varepsilon}^{-1}w$.

Метод интерпретатора. Кроме методов регуляризации, которые можно отнести к разряду объективных (не забывая, впрочем, что сама постановка каждой из вариационных задач, выбор конкретной нормы и параметра регуляризации содержат, особенно в практических расчетах, определенный субъективный элемент), существует и давно применяется чисто субъективный *метод интерпретатора*. Он состоит в том, что опытный специалист (интерпретатор) подбирает функцию $v(0, x)$, как удовлетворяющую условию типа (2), так и обладающую рядом свойств, ограничивающих выбор. Эти свойства часто даже явно не формулируются: просто интерпретатор знает, какие функции $v(0, x)$ бывают в данной задаче, а каких быть не может.

Фактическое решение сложных прикладных обратных задач осуществляется, разумеется, комбинацией объективного и субъективного методов: получаемые методом регуляризации решения подвергаются тщательному качественному анализу, и в случае, если решение оказывается по тем или иным причинам неудов-

летворительными, исследуется другое, полученное, например, с другим значением параметра регуляризации. Этот этап взаимодействия вычислителя с заказавшим расчет специалистом чрезвычайно плодотворен: он приводит к уточнению постановки задачи, заставляет специалиста четко и определенно формулировать требования, предъявляемые к решению. При этом те свойства, которые казались сами собой разумеющимися при начальной формулировке задачи и не требующими специального явного включения в ее условия, после того как получено решение с очевидными (для специалиста с его интуицией и неформализованными знаниями) дефектами, выявляются, и соответствующие условия легче поддаются формализации и явному включению в задачу. Этот этап уточнения постановки задачи может состоять из несколько подобных «итераций». Однако тут возникает определенное противоречие между той априорной информацией о решении, которой располагает специалист, и теми средствами, которыми располагает вычислитель для удовлетворения поставленных требований. Так, специалист может утверждать, например, что $u(x) \geq 0$, что $u(x) \leq c$, что $u(x)$ не должна иметь слишком частых колебаний, однако разрывы не исключаются, и тому подобное. Между тем вычислитель располагает обычно лишь величиной ϵ параметра регуляризации и теоремой о том, что если требуемое решение существует, то его можно найти при подходящем значении этого параметра. Заметим, что перечисленные выше способы регуляризации сформулированы в достаточно общей форме, и при соответствующей конкретизации, например, норм, входящих в условие вариационной задачи, могут учитывать разнообразные требования. Однако, есть и сложившаяся вычислительная практика, в которой, как правило, используются нормы в пространстве L_2 . Наиболее освоенные и простые с точки зрения использования стандартных средств вычислений, они, к сожалению, плохо приспособлены для учета упоминавшихся качественных данных о решении. Эти нормы удобны, когда искомое решение лежит в некотором линейном пространстве, однако качественная информация о решении выделяет обычно не линейное пространство, а, например, выпуклый конус, или выпуклое тело. Так, часто используемое условие

$$\left\| \frac{du}{dx} \right\|^2 = \int_0^1 \left(\frac{du}{dx} \right)^2 dx \leq A \quad \text{хорошо приспособлено для подавления}$$

высокочастотных колебаний, но одновременно исключает возможные в принципе разрывы в $u(x)$. Метод квазиобращения [45] так же плохо приспособлен для получения, например, положительных решений. Правда, в [45] (гл. 6) намечаются пути решения обратных задач с учетом каких-то условий на $u(x)$, однако технически это предлагается сделать, выбирая $u(x)$ в виде линейной комбинации из некоторого числа функций, каждая из которых

обладает нужным свойством. Такой подход приводит к тому, что решение ищется в некотором линейном пространстве, а это не очень удобное средство учета качественных требований: так, положительные функции образуют выпуклый конус, но не линейное пространство. Методы решения вариационных задач, рассматриваемые в настоящей книге, как раз и ориентированы на отыскание функций с подобного рода качественными ограничениями. Разумеется, это требует привлечения более сложных вычислительных средств, но задачи стоят этого. Ниже все это будет проиллюстрировано решением модельной обратной задачи теплопроводности. С чисто теоретической точки зрения ничего нового не предлагается, так как наш подход укладывается в давно известную общую схему такого, например, типа: найти $u(\cdot)$ решением задачи

$$\min_u \|Ru - w\| \quad \text{при условии} \quad u(\cdot) \in U.$$

Речь идет лишь о том, чтобы в условие $u(\cdot) \in U$ включить возможно большее число ограничений, причем в той форме, которая наиболее полно и непосредственно учитывает имеющуюся качественную информацию об искомом решении.

Сведем обратную задачу к вариационной. Начнем с решения следующей задачи «оптимального управления». Найти $v(0, x)$, $0 \leq x \leq 1$ из условий:

$$\begin{aligned} \text{I.} \quad & \min_{v(\cdot)} \int_0^1 q(x) v(0, x) dx, \\ \text{II.} \quad & \max_x |v(T, x) - w(x)| \leq \delta, \\ \text{III.} \quad & \text{var } v(0, \cdot) \leq W. \end{aligned} \tag{5}$$

Второе условие имеет очевидный содержательный смысл, третье является примером возможной формы включения содержательных требований: ограничив вариацию искомой функции $v(0, x)$ некоторым числом W , мы, не исключая, например, разрывных решений, можем запретить слишком большие и частые осцилляции. Особую роль играет первое условие, оно носит совсем произвольный характер, мы еще разъясним его назначение. Рассматривается вариант с $T=0,01$. В качестве $w(x)$ возьмем функцию, полученную следующим образом. Положим

$$v(0, x) = \begin{cases} 0 & \text{при } x < 0,3 \text{ или } x > 0,5, \\ 1 & \text{при } 0,3 < x < 0,5, \end{cases} \tag{6}$$

и решив прямую задачу теплопроводности (1), найдем $v(T, x)$. Таким образом, (6) есть искомое решение. На рис. 61 изображены $v(0, x)$ и, $w(x)$. Число δ («точность измерения» w) будем считать не так уж ма-

лым: $\delta=0,015$ (при характерной величине $w(x) \approx 0,3$). Вариация точного решения равна 2, мы зададим сначала $W=3,2$. Теперь обратимся к условию I. Оно введено по следующим соображениям. Одним из основных вопросов, возникающих в связи с постановкой обратной задачи, является вопрос о функциональном поперечнике множества функций $v(0, x)$, удовлетворяющих условиям II и III). Другими словами, о том, насколько могут различаться функции, удовлетворяющие этим условиям. Это, в сущности, вопрос о гарантированной точности решения. Если учесть, что выше была поставлена модельная задача, и включение других априорных сведений о $v(0, x)$ приведет к осложнениям, исключающим сколько-нибудь реальную теоретическую оценку поперечника, в нашем распоряжении остаются лишь вычислительные эксперименты. Хотя они и не дадут точного ответа, но все-таки какие-то сведения на этот счет можно получить. Вводя условие I, мы имеем в виду, решив задачу при разных $q(x)$, получать разные $v(0, x)$, лежащие на границе множества, выделяемого условиями II, III. Сравнивая их, мы сможем составить хоть какое-то представление о том, с какой точностью условия II, III (и им подобные) определяют $v(0, x)$. В принципе можно было бы поставить и задачу определения двух функций $v(0, x)$ и $v^*(0, x)$, каждая из которых удовлетворяет II, III и, кроме того, которые доставляют

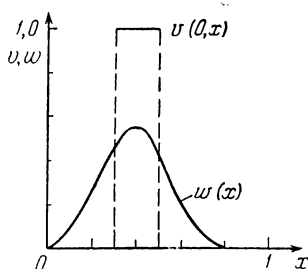


Рис. 61.

$$\max_{v, v^*} \|v(0, \cdot) - v^*(0, \cdot)\|.$$

Есть и другая причина для того, чтобы ввести условие I и решать задачу многократно, с разными $q(x)$. Если мы каким-то образом найдем одну из функций, удовлетворяющих II, III, то в ней будет как полезная информация, так и случайная, связанная с неоднозначностью. Можно ожидать, что в разных решениях (соответствующих разным $q(x)$) будут какие-то устойчивые элементы, общие для всех решений, и неустойчивые, характерные для данной функции $q(x)$. Имея несколько таких решений, можно попытаться отделить случайное от закономерного. Впрочем, лучше будет вернуться к этому вопросу, уже имея результаты вычислений. Для моделирования условия III используем замену искомого функции $v(0, x)$ на функцию $u(x)$, связанную с v уравнением

$$\frac{dv(0, x)}{dx} = u(x); \quad v(0, x) = 0, \quad (7)$$

а вместо $\text{var } v(0, \cdot) \leq W$ поставим фактически эквивалентное условие

$$F[u(\cdot)] \equiv \int_0^1 |u(x)| dx \leq W. \quad (8)$$

С подобными функционалами мы уже имели дело в § 34. Как и там, здесь нам придется вместо $u(x)$ ввести две функции:

$$u^+(x) \geq 0; \quad u^-(x) \leq 0; \quad u(x) = u^+(x) + u^-(x),$$

а вместо (8) поставим условие

$$\int_0^1 \{u^+(x) - u^-(x)\} dx \leq W.$$

Нельзя гарантировать, что в решении в каждой точке только одна из функций (u^+ или u^-) отлична от нуля, но заведомо $u^+(x) - u^-(x) \geq |u^+(x) + u^-(x)|$, и соотношение (8) будет выполнено. Таким образом, приходим к задаче: найти управление $\{u^+(\cdot), u^-(\cdot)\}$, минимизирующее функционал $F_0[u^+(\cdot), u^-(\cdot)]$ при условиях $F_i[u^+(\cdot), u^-(\cdot)] = 0 (\leq 0)$, $i=1, 2, \dots$. Осталось только определить функционалы

$$F_0[u^+(\cdot), u^-(\cdot)] \equiv \int_0^1 \{u^+ + u^-\} q(x) dx,$$

$$F_1[u^+(\cdot), u^-(\cdot)] \equiv \int_0^1 \{u^+ - u^-\} dx - W \quad (\leq 0),$$

$$F_2[u^+(\cdot), u^-(\cdot)] \equiv v(0, 1) = \int_0^1 \{u^+ + u^-\} dx \quad (=0),$$

$$F_3[u^+(\cdot), u^-(\cdot)] \equiv \max_x |v(T, x) - w(x)| \quad (\leq \delta).$$

Управление $\{u^+(\cdot), u^-(\cdot)\}$ определяет $v(T)$ решением уравнения (7), а затем прямого уравнения теплопроводности (1). Аппроксимация функционала F_3 осуществлялась следующим образом: интервал $0 \leq x \leq 1$ разбивался на K равных частей, на каждой k -й части находилась точка x^k , доставляющая $\max_x |v(T, x) - w(x)|$,

и в вычислениях полагалось

$$F_3 \approx \max_{k=1, \dots, K} |v(T, x^k) - w(x^k)|.$$

Разумеется, точки аппроксимации выбирались на каждой варьированной траектории. В остальном решение вариационной задачи

осуществлялось в соответствии со схемой §§ 19—21 (см. также § 34). Управление u^+ , $u^- (x)$ аппроксимировалось кусочно постоянной функцией на сетке с шагом $\Delta x = 10^{-2}$; вариации u не превосходили заданной величины S . Основные затраты машинного времени на одну итерацию были связаны со следующими вычислениями:

- 1) решение прямого уравнения теплопроводности (1);
- 2) K -кратное решение сопряженного уравнения $\psi_t = -\psi_{xx}$ с начальными данными при $t=T$; $\psi(T, x) = \delta(x-x^k)$ (δ -функция в расчетах аппроксимировалась сеточной функцией, равной $1/\Delta x$ в точке x^k и нулю в остальных);
- 3) решение задачи линейного программирования размером $200 \times (3+K)$.

Задача содержит определенные трудности. Ведь искомая функция $v(0, x)$ имеет два разрыва, в ее разложении в ряд Фурье коэффициенты убывают не очень быстро, и хорошее восстановление $v(0, x)$ затруднено тем, что в $v(T, x)$ соответствующие гармоники уже теряются в ошибках $\sim \delta$. Замена искомой функции $v(0, x)$ на $u(x)$ имеет и положительные, и отрицательные следствия. Положительным является своеобразный эффект регуляризации: так как мы ограничимся относительно небольшим числом вариаций функции $u(x)$ на величины $|\delta u(x)| \leq S$, то получить очень уж негладкую функцию $v(0, x)$ не удастся. С другой стороны, эта замена затрудняет и получение разрывов в $v(0, x)$: ведь это требует построения в $u(x)$ каких-то аппроксимаций δ -функций.

Перейдем к анализу результатов вычислений.

Первый расчет: $S=0,25$, $K=5$, $W=3,2$, $\delta=0,015$, $q(x)=1$. В качестве исходной функции бралась функция, равная 2 при $0 \leq x \leq 0,5$ и -2 при $0,5 < x < 1$. В этом расчете первые 23 итерации составляет решение терминальной задачи: находится функция $u(x)$, удовлетворяющая всем условиям задачи. Затем начинается эволюция $u(x)$ с целью минимизации $F_0[u(\cdot)]$. Сама по себе величина F_0 нас не интересует, поэтому соответствующие данные не приводятся. Найденная в конце расчета функция $v(0, x)$ показана на рис. 62 (I). Заметим, что попытка проведения первого расчета с $K=3$ оказалась неудачной — трех точек аппроксимации недостаточно для того, чтобы обеспечить условие $\max |v(T, x) - w(x)| \leq \delta$.

Второй расчет отличался от первого только величиной $W=2,2$. Результат представлен на рис. 62 (II).

Третий расчет отличался от первого величиной $W=3,0$ и функцией $q(x)=-1$. Результат представлен на рис. 62 (III).

Четвертый расчет отличался от третьего только тем, что требуемая точность совпадения $\max_x |v(T, x) - w(x)| \leq \delta$ была увеличена: $\delta=0,0075$, а не 0,015, как в первых трех расчетах. Результат представлен на рис. 62 (IV). Повышение точности потребовало

увеличения числа точек аппроксимации K с 5 до 7. Это очень важное обстоятельство: при дальнейшем увеличении точности задача становится все более трудной.

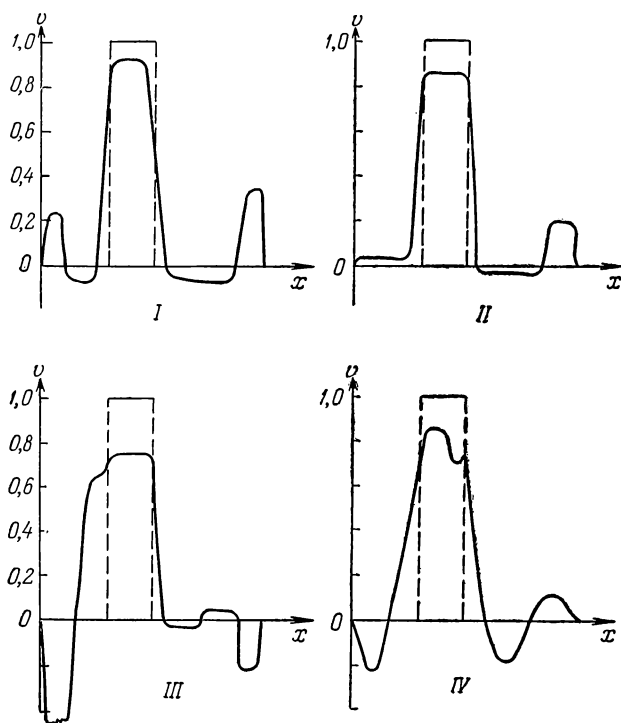


Рис. 62.

Следующие расчеты имели целью показать, что дает подключение к условиям задачи новой качественной информации. Было поставлено условие:

$$v(0, x) \geq 0 \quad \text{при} \quad 0 \leq x \leq 1. \quad (9)$$

Оно аппроксимировалось всего двумя точками, т. е. вместо (9) в расчетах было два условия: $v(0, x') \geq 0$, $v(0, x'') \geq 0$; точки аппроксимации x' , x'' выбирались в местах наибольшего нарушения (9). Точность выполнения (9) была не очень высокой, но этого было достаточно.

Пятый расчет: $q(x) = -1$, $W = 3$, $\delta = 0,015$. Результат представлен на рис. 63 (I).

Шестой расчет отличался от пятого только величиной $W = 2,2$. Результат показан на рис. 63 (II).

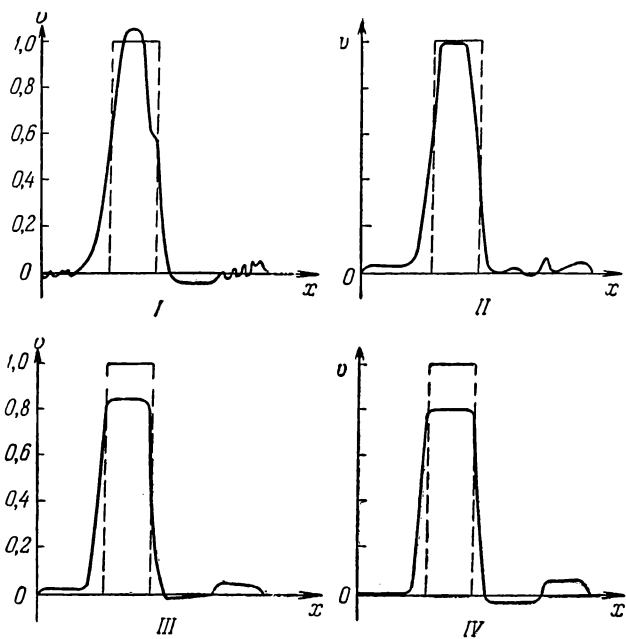


Рис. 63.

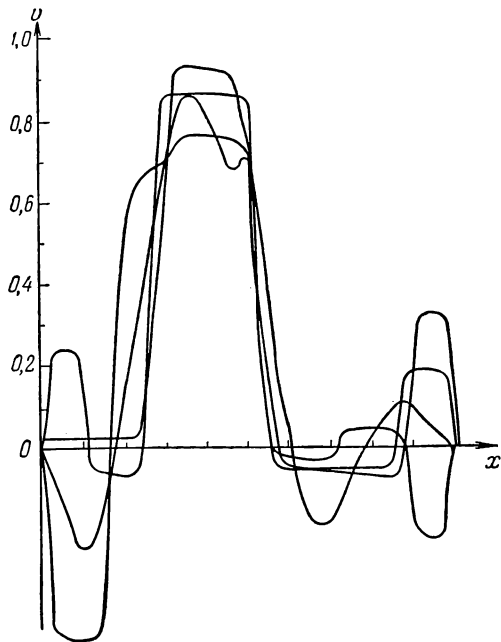


Рис. 64.

Седьмой расчет (рис. 63 (III)) имел целью проверить, как влияет на решение постепенное изменение условия $\text{var } v(0, \cdot) \leq W$. Ведь в реальных задачах такие ограничения не очень строги, так как значение W известно неточно. В этом расчете $W=1,8$, т. е. меньше, чем в искомом решении. Можно ожидать, что уменьшение W должно прежде всего сказаться на случайных деталях численного решения. Расчет подтвердил это предположение.

Восьмой расчет отличался от седьмого только тем, что условие $v(0, x) \geq 0$ было снято. Результат представлен на рис. 63, IV. Он также подтверждает соображения о том, что, варьируя те параметры задачи (W в данном случае), которые по смыслу ее постановки не очень точны, можно в известной мере отделять существенные детали численного решения от случайных. Разумеется, делать это нужно очень осторожно.

На рис. 64 показаны результаты расчетов 1—4, нанесенные на один график. Этим иллюстрируется возможность анализировать результаты, сравнивая различные, формально равноправные, решения обратной задачи. Заметим, что каждый из представленных расчетов занимал ≈ 45 минут на БЭСМ-6.

Г Л А В А IV

СТАНДАРТНЫЕ АЛГОРИТМЫ

§ 42. Основные свойства выпуклых множеств

Теория вариационных задач тесно связана с теорией выпуклых тел. Выпуклые тела возникают и в алгоритмах приближенного решения. В этом параграфе приводятся основные сведения о выпуклых телах, используемые в приближенных методах. В основном мы будем иметь дело с выпуклыми телами в конечномерных пространствах.

О п р е д е л е н и е 1. Множество точек Q называется *выпуклым*, если вместе с любой парой точек q_1, q_2 множество Q содержит

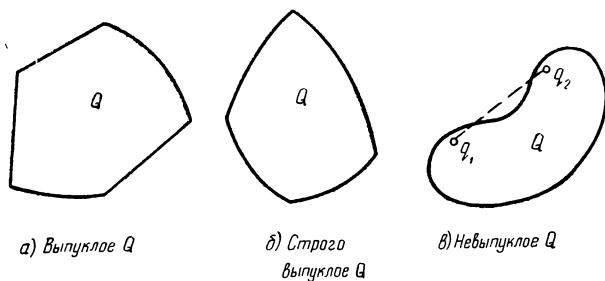


Рис. 65.

и соединяющий их отрезок прямой. Другими словами, если $q_1 \in Q$ и $q_2 \in Q$, то все точки $s \cdot q_1 + (1-s)q_2 \in Q$ при любом $0 \leq s \leq 1$.

О п р е д е л е н и е 2. Выпуклое тело называется *строго выпуклым*, если все точки отрезка (за исключением, быть может, его концов), соединяющего точки $q_1 \in Q, q_2 \in Q$, лежат строго внутри Q . Иначе свойство строгой выпуклости можно сформулировать так: пусть $q_1 \in Q$ и $q_2 \in Q$, и s — некоторое число $0 < s < 1$; тогда можно построить сферу некоторого радиуса $\varepsilon > 0$ (своего для каждого s) с центром в точке $sq_1 + (1-s)q_2$, целиком лежащую в Q .

Полезно иметь в виду и следующий простой признак, различающий строго выпуклые тела от просто выпуклых: граница строго выпуклого тела не может содержать отрезки прямых или куски плоскостей (рис. 65).

Нас в основном будут интересовать замкнутые выпуклые тела. Границу такого тела Q будем обозначать ∂Q , причем $\partial Q \in Q$ в силу замкнутости. Важным для дальнейшего является следующий объект, обобщающий привычное понятие касательной плоскости.

О п р е д е л е н и е 3. Гиперплоскость G , проходящую через точку $q_1 \in \partial Q$ ортогонально некоторому вектору g , будем называть *опорной* к Q в точке q_1 , если

$$(q - q_1, g) \geq 0 \text{ для всех } q \in Q.$$

Другими словами, выпуклое тело Q лежит целиком в одном из двух полупространств, на которые G делит все пространство (рис. 66, а).

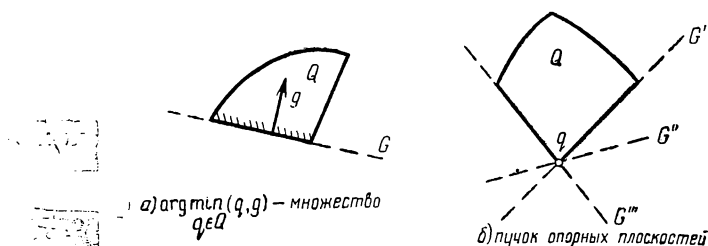


рис. 66.

Если тело Q замкнуто и ограничено, то каждому вектору g соответствует (быть может, не единственная) точка q_1 , в которой g определяет опорную к Q гиперплоскость. Эта точка q_1 есть решение простейшей задачи выпуклого программирования:

$$\min_{q \in Q} (q, g). \quad (1)$$

Так как Q замкнуто и ограничено, существует по крайней мере одна точка q_1 , в которой достигается этот минимум:

$$(q_1, g) = \min_{q \in Q} (q, g), \text{ т. е. } (q - q_1, g) \geq 0 \text{ для всех } q \in Q. \quad (2)$$

В силу линейности по q минимизируемой функции (q, g) , $q_1 \in \partial Q$. Совокупность всех точек ∂Q , в которых достигается минимум (q, g) , обозначают символом

$$\arg \min_{q \in Q} (q, g). \quad (3)$$

Очевидно, что все точки этого множества лежат в гиперплоскости G . Важным для дальнейшего является следующий простой факт.

Теорема 1. Если Q — строго выпуклое замкнутое ограниченное тело, то $\min_{q \in Q} (q, g)$ при любом g достигается в единственной точке $q(g)$:

$$(q(g), g) = \min_{q \in Q} (q, g) \text{ или } q(g) = \arg \min_{q \in Q} (q, g). \quad (4)$$

Следующая важная теорема об отделимости выпуклых тел существенно используется как в теоретическом анализе, так и в вычислениях.

Т е о р е м а 2. Пусть Q_1 и Q_2 — выпуклые замкнутые ограниченные тела, не имеющие общих точек ($Q_1 \cap Q_2 = \emptyset$). Тогда существует гиперплоскость G (проходящая через некоторую точку q^* ортогонально вектору g), строго разделяющая тела Q_1 и Q_2 в том смысле, что

$$\begin{aligned} (q' - q^*, g) &< 0 \quad \text{для всех } q' \in Q_1, \\ (q'' - q^*, g) &> 0 \quad \text{для всех } q'' \in Q_2 \end{aligned} \quad (5)$$

(иначе говоря, G делит пространство на два полупространства, и Q_1 лежит целиком в одном из них, а Q_2 — в другом).

Д о к а з а т е л ь с т в о. Определим точки $q_1 \in Q_1$ и $q_2 \in Q_2$ решением (не обязательно единственным) следующей задачи (рис. 67):

$$\min_{q' \in Q_1, q'' \in Q_2} \|q' - q''\| = \|q_1 - q_2\|. \quad (6)$$

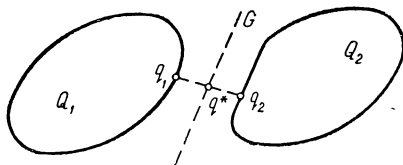


Рис. 67.

Решение этой задачи существует в силу замкнутости и ограниченности Q_1 и Q_2 . Положим $q^* = \frac{1}{2}(q_1 + q_2)$ и $g = q_2 - q_1$. Докажем неравенства (5). Пусть первое из них неверно, и в Q_1 существует точка q' , для которой $(q' - q^*, g) \geq 0$. Покажем, что в этом случае при перемещении точки q вдоль соединяющего q_1 с q' отрезка (а он, в силу выпуклости Q_1 , целиком состоит из точек Q_1), расстояние этой движущейся точки до q_2 (а тем более и до Q_2) строго убывает (в окрестности q_1). Итак, введем «движущуюся» точку $q(s) = q_1 + s(q' - q_1)$, $0 \leq s \leq 1$, $q(s) \in Q_1$ и вычислим (при $s=0$)

$$\begin{aligned} \frac{d}{ds} \|q(s) - q_2\|^2 &= \frac{d}{ds} (q_1 + s(q' - q_1) - q_2, q_1 + s(q' - q_1) - q_2) \Big|_{s=0} = \\ &= 2(q_1 - q_2, q' - q_1) = -2 \left(g, q' - \frac{q_2 + q_1}{2} + \frac{q_2 - q_1}{2} \right) = \\ &= -2(g, q' - q^*) - (g, g) < 0, \end{aligned}$$

так как мы предположили, что $(g, q' - q^*) > 0$. Точно так же проверяется и второе из неравенств (5). Полученное противоречие с определением точек q_1 и q_2 (6) доказывает теорему.

З а м е ч а н и е. Почти без всяких изменений проходит доказательство теоремы 2 в случае, когда одно из тел Q — неограничено.

Теорема 1 утверждает, что каждому вектору g соответствует по крайней мере одна точка $q(g)$ границы выпуклого множества Q , в которой g определяет опорную к Q гиперплоскость G . Наоборот,

в каждой точке $q \in \partial Q$ может быть построена опорная к Q гиперплоскость G .

Т е о р е м а 3. Пусть Q — выпуклое множество, $q^* \in \partial Q$. Тогда существует вектор g такой, что $(q - q^*, g) \geq 0$ для всех $q \in Q$.

Д о к а з а т е л ь с т в о. Так как q^* — точка границы Q , то существует вектор e такой, что луч $q^* + se$, $s > 0$ целиком лежит вне Q . Выберем последовательность чисел $s_1 > s_2 > \dots > s_i > \dots \rightarrow 0$ и образуем последовательность точек $q_i = q^* + s_i e \notin Q$, $i = 1, 2, \dots$. Каждая точка q_i есть выпуклое множество, и, по теореме об отделимости, для каждого i существует гиперплоскость G_i , определяемая нормированным вектором g_i , причем

$$(q - q_i, g_i) > 0 \text{ для всех } q \in Q.$$

Последовательность q_i сходится к точке q^* , из последовательности g_1, g_2, \dots можно выбрать сходящуюся к некоторому вектору g подпоследовательность (чтобы не усложнять обозначений, будем считать $g_i \rightarrow g$). Переходя к пределу по $i \rightarrow \infty$, получим для каждой точки $q \in Q$ неравенство

$$(q - q^*, g) \geq 0.$$

Теорема доказана.

Рассмотрим типичную задачу математического программирования: найти точку u n -мерного пространства, удовлетворяющую условиям:

$$1) \quad u \in U, \text{ где } U \text{ — ограниченная замкнутая область;} \quad (7)$$

$$2) \quad F_i(u) = 0, \quad i = 1, 2, \dots, m, \quad (8)$$

и минимизирующую значение функции $F_0(u)$, т. е. найти

$$\min_{u \in U} F_0(u) \quad (9)$$

при условиях (7), (8).

Введем $(m+1)$ -мерное пространство точек F и в нем Q — образ области U , определяемый отображением

$$F(u) = \{F_0(u), F_1(u), \dots, F_m(u)\}. \quad (10)$$

Все функции $F_i(u)$ будем считать достаточно гладкими. Мы будем решать эту задачу, приняв следующее

П р е д п о л о ж е н и е. Образ U в отображении $F(u)$ есть строго выпуклое замкнутое ограниченное множество.

В этом случае задача на условный экстремум (9) может быть сформулирована в виде следующей задачи строго выпуклого программирования: в строго выпуклом ограниченном замкнутом множестве Q нужно найти точку вида λe с наименьшим значением λ (здесь $e = \{1, 0, 0, \dots, 0\}$), т. е.

$$\min \lambda \text{ при условии } \lambda e \in Q. \quad (11)$$

Не следует забывать, что Q задано нам отображением $F(u)$, $u \in U$; для образа U в этом отображении мы будем использовать обозначение $Q = F(U)$. Разумеется, нас будет интересовать как значение λ в решении задачи (11), которое обозначим Λ , так и прообраз u^* точки Λe : $\Lambda e = F(u^*)$ (или один из этих прообразов, если задача (9) имеет неединственное решение). Задача (9) связана с определенным вектором e , однако в следующем ниже изложении можно будет считать вектор e произвольным; разумеется, при этом задача (11) уже не будет соответствовать задаче (9). Целью дальнейшего является сведение задачи на условный экстремум (9) к задачам на безусловный экстремум для некоторых новых функций. Введем множество $(m+1)$ -мерных векторов $g = \{g_0, g_1, \dots, g_m\}$, нормированных условием $(g, e) = 1$.

Т е о р е м а 4. *Задача строго выпуклого программирования (11) эквивалентна задаче*

$$\max_g \{ \min_{q \in Q} (q, g) \}. \quad (12)$$

Эта теорема сводит задачу на условный экстремум к суперпозиции задач на безусловный экстремум. В самом деле, определим функцию $R(g)$:

$$R(g) \equiv \min_{q \in Q} (q, g) = \min_{u \in U} \sum_{i=0}^m g_i F_i(u). \quad (13)$$

Если иметь в виду приложение теоремы 4, то не следует забывать, что в общем случае функция $R(g)$ определяется некоторыми алгоритмами приближенного отыскания минимума функции $\sum g_i F_i(u)$. Так как мы предположили все F_i достаточно гладкими, то в общем случае $R(g)$ может определяться, например, алгоритмом спуска по градиенту. Есть два частных случая, когда вычисление $R(g)$ может быть не очень сложным и осуществляется, в принципе точно в результате конечного числа операций. Это случаи линейных и квадратичных зависимостей $F_i(u)$ и не очень сложных областей U , определяемых, например, условиями вида

$$U: |u_i| \leq 1, \quad i = 1, 2, \dots, n. \quad (14)$$

Подобные задачи встречаются в качестве элементов в алгоритмах решения более сложных задач и поэтому заслуживают внимания.

Так или иначе, функция $R(g)$ определена, и теорема 4 сводит задачу (11) (или (9)) к задаче

$$\max_{(g, e)=1} R(g). \quad (15)$$

Это, по существу, задача на безусловный экстремум, так как условие нормировки $(g, e) = 1$ легко учитывается как при аналитическом решении этой задачи (если оно возможно), так и при численном методе подъема по градиенту.

Доказательство теоремы 4 следует из двух простых лемм.

Лемма 1. *Каков бы ни был вектор g , нормированный условием $(g, e) = 1$,*

$$R(g) \leq \Lambda. \quad (16)$$

В самом деле, $\Lambda e \in Q$, и

$$R(g) = \min_{q \in Q} (g, q) \leq (g, \Lambda e) = \Lambda.$$

Следовательно, $\max R(g) \leq \Lambda$.

Лемма 2. *$\max_{g \in Q} R(g) \geq \Lambda$, и достигается этот максимум на векторе g^* , определяющем опорную к Q в точке Λe гиперплоскость.*

В самом деле, пусть g^* — вектор, ортогональный опорной к Q в точке $\Lambda e \in \partial Q$ гиперплоскости G^* . Тогда

$$R(g^*) = (\Lambda e, g^*) = \Lambda, \quad (17)$$

следовательно, учитывая (16), получим соотношение

$$\max_{(g, e)=1} R(g) = \max_{(g, e)=1} \min_{q \in Q} (g, q) = \Lambda. \quad (18)$$

Пусть теперь \tilde{g} — произвольный вектор из множества $\arg \max_{(g, e)=1} R(g)$,

т. е. $\min_{q \in Q} (q, \tilde{g}) = \Lambda$, и пусть \tilde{q} — произвольная точка из множества $\arg \min_{q \in Q} (q, \tilde{g})$. Тогда точка Λe лежит в гиперплоскости,

проходящей через \tilde{q} ортогонально \tilde{g} ; таким образом, эта гиперплоскость является опорной к Q и в точке Λe . В самом деле, для всех $q \in Q$ $(q - \tilde{q}, \tilde{g}) \geq 0$, а для $q = \Lambda e \in Q$ имеем

$$(\Lambda e - \tilde{q}, \tilde{g}) = (\Lambda e, \tilde{g}) - (\tilde{q}, \tilde{g}) = \Lambda - \Lambda = 0.$$

Теорема доказана.

Заметим, что вектор g , решающий задачу $\max R(g)$, может не быть единственным. Далее, теорема о том, что $\Lambda = \max_{g \in Q} \min_{q \in Q} (g, q)$, верна и для просто выпуклых множеств; строгая выпуклость не обязательна. Однако в этом случае решение задачи $\max_{(g, e)=1} \min_{q \in Q} (g, q)$ не обязательно даст нам решение исходной задачи математического программирования: искомая точка Λe может быть лишь одной из множества точек ∂Q , в которых достигается $\min_{q \in Q} (q, g^*)$. Лишь в случае строго выпуклого множества Q гарантируется эквивалентность задач (12) и (11). Возможные ситуации иллюстрирует рис. 68.

Задача на условный экстремум (7)–(9), которую при определенных предположениях удалось свести к суперпозиции задач

на безусловный экстремум (12), очень часто возникает в приложениях. Настоящая книга посвящена, в сущности, некоторому конкретному классу подобных задач. Поэтому нам стоит подробнее разобраться в этом вопросе с вычислительной точки зрения. Прежде всего, функция $R(g)$ в общем случае определена лишь некоторым вычислительным процессом — алгоритмом решения задачи $\min_{q \in Q} (q, g)$. Подобные алгоритмы обычно дают лишь приближенное значение $R(g)$, причем достижение высокой точности связано с большими затратами машинного времени. Затем возникает задача нахождения $\max R$, для решения которой очень

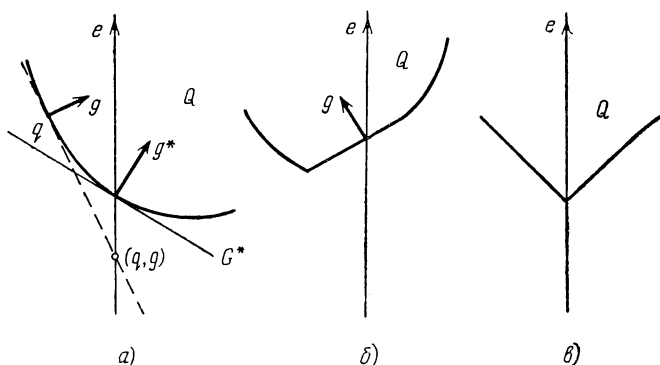


Рис. 68.

полезно уметь вычислять производные $\partial R(g)/\partial g$. Разумеется, в нашем распоряжении всегда остается метод численного дифференцирования, однако к нему следует прибегать лишь в крайнем случае, когда остальные способы по каким-либо причинам неприемлемы. Ведь численное дифференцирование в данном случае состоит в решении $(m+1)$ -й задачи поиска $\min_{q \in Q} (q, g)$ — для вектора $g = \{1, g_1, \dots, g_m\}$, затем для возмущенных векторов $\{1, g_1 + \Delta, g_2, \dots, g_m\}, \dots, \{1, g_1, \dots, g_m + \Delta\}$. Кроме того, $R(g)$ вычисляется с какими-то ошибками δR , а при численном дифференцировании они возрастают в Δ^{-1} раз. Заметим еще, что численное дифференцирование в силу своей крайней простоты и универсальности часто оставляет в тени очень важный вопрос о том, существуют ли вычисляемые производные. При этом наиболее доступный способ практической проверки дифференцируемости функции, состоящий в вычислении и анализе величин

$$\frac{R(g + \Delta l) - R(g)}{\Delta}, \quad \frac{R\left(g + \frac{\Delta}{2} l\right) - R(g)}{\Delta/2}, \dots,$$

может ввести в заблуждение: даже если последовательно вычисляемые приближенные значения производной обнаруживают явную тенденцию к установлению (мы считаем, что шаг дифференцирования $\Delta/2^k$ еще не настолько мал, чтобы начали сказываться ошибки в вычислении R), это свидетельствует лишь о существовании у функции $R(g)$ производной по направлению l . Есть еще одно обстоятельство, которое оправдывает проявление осторожности в данном вопросе, даже если все функции $F_i(u)$, входящие в первичную постановку задачи (7)–(9), сколь угодно гладкие. Дело в том, что операция тах может из сколь угодно гладких функций сделать функции недифференцируемые.

Однако рассматриваемый нами случай строго выпуклого программирования в этом отношении вполне благополучен: функция $R(g)$ дифференцируема. Более того, ее производная $\partial R/\partial g$ вычисляется достаточно просто, и нет необходимости прибегать к численному дифференцированию.

Теорема 5. *Для строго выпуклой замкнутой ограниченной области Q функция $R(g) \equiv \min_{q \in Q} (q, g)$ (ее иногда называют опорной функцией) дифференцируема, и ее производная вычисляется по формуле*

$$\frac{\partial R(g)}{\partial g} = q(g), \text{ где } q(g) = \arg \min_{q \in Q} (q, g). \quad (19)$$

Доказательство. Проверим, что для любого вектора z

$$\lim_{s \rightarrow 0} \frac{R(g + sz) - R(g)}{s} = (q(g), z). \quad (20)$$

Используем следующие обозначения: $g_s = g + sz$, $q_s = q(g_s)$, $r_s = q_s - q_0$. Тогда

$$\begin{aligned} \frac{R(g_s) - R(g_0)}{s} &= \frac{(g_s, q_s) - (g_0, q_0)}{s} = \frac{(g_0 + sz, q_0 + r_s) - (g_0, q_0)}{s} = \\ &= (q_0, z) + \frac{1}{s} [(r_s, g_0) + s(z, r_s)]. \end{aligned}$$

Для доказательства достаточно показать, что

$$\lim_{s \rightarrow 0} \frac{1}{s} [(r_s, g_0) + s(z, r_s)] = 0,$$

Используем следующие неравенства:

$$\begin{aligned} (g_0, q_0) &\leq (g_0, q_0 + r_s), \text{ сл. } (g_0, r_s) \geq 0; \\ (g_s, q_s) &\leq (g_s, q_0), \text{ т. е. } (g_0 + sz, q_0 + r_s) \leq (g_0 + sz, q_0), \end{aligned}$$

откуда следует $(g_0, r_s) + s(z, r_s) \leq 0$.

Таким образом, используя равномерную оценку $|(z, r_s)| \leq C_1$, являющуюся очевидным следствием ограниченности Q , получаем

$$0 \leq (g_0, r_s) \leq -s(z, r_s) \leq C_1 s. \quad (21)$$

Отсюда следует, что $\lim_{s \rightarrow 0} (g_0, r_s) = 0$. Покажем теперь, что и $\lim_{s \rightarrow 0} \|r_s\| = 0$. Пусть это не так, и для последовательности $s_1 > s_2 > \dots \rightarrow 0$, соответствующие величины $\|r_{s_i}\| \geq a > 0$. Тогда из совокупности векторов r_{s_i} можно выбрать сходящуюся подпоследовательность (для которой мы не станем вводить особых обозначений); пусть r — ее предел. Очевидно, $\|r\| \geq a$. Далее, все точки $q_0 + r_{s_i} \in Q$, следовательно, и $q_0 + r \in Q$, а переходя к пределу $s_i \rightarrow 0$ в (21), получим $(g_0, r) = 0$, т. е. $(g_0, q_0 + r) = (g_0, q_0) = \min_{q \in Q} (g_0, q)$. Таким образом, минимум (g_0, q) достигается по крайней мере в двух разных точках q , что несовместимо со строгой выпуклостью Q . Итак, $\|r_s\| \rightarrow 0$ при $s \rightarrow 0$. Теперь можно утверждать, что $-s(g_0, r_s) = o(s)$; то же самое в силу (21) относится и к (g_0, r_s) . Этим доказательство теоремы завершено.

Теперь осталось построить метод нахождения $\max_g R(g)$, учитывая еще условие нормировки. Итак, имеем задачу

$$\max_g R(g) \text{ при условии } (g, e) = 1.$$

Следуя правилу Лагранжа, образуем функцию $\Lambda(g, \lambda) \equiv R(g) + \lambda(g, e)$ и вычислим ее градиент

$$\frac{\partial \Lambda}{\partial g} = \frac{\partial R}{\partial g} + \lambda \frac{\partial}{\partial g} (g, e) = q(g) + \lambda e.$$

Метод подъема по условному градиенту состоит в переходе от вектора g к вектору $g_s = g + s \frac{\partial \Lambda}{\partial g} = g + sq(g) + s\lambda e$, а множитель λ выбирается таким, чтобы условие нормировки не было нарушено:

$$1 = (g_s, e) = (g + sq + \lambda se, e) = 1 + s(q, e) + \lambda s(e, e),$$

откуда

$$\lambda = - \frac{(q(g), e)}{(e, e)}.$$

Шаг подъема s определяется, например, решением одномерной задачи $\max_s R(g_s)$.

§ 43. Метод Ньютона

Предназначенный для решения (вернее, для уточнения грубых приближений к решению) систем нелинейных уравнений, метод хорошо известен. Известна и высокая скорость его сходимости. Итак, нужно найти решение системы уравнений

$$f(x) = 0 \quad (1)$$

(здесь $f = \{f^1, f^2, \dots, f^n\}$, $x = \{x_1, x_2, \dots, x_n\}$). Точнее, нужно, исходя из имеющегося приближения x^0 , найти достаточно точное ближайшее к x^0 решение системы (1) — точку x^* .

Вычислительная схема метода основана на предположении о достаточной малости отклонения $\delta x^0 = x^0 - x^*$ и о возможности пренебречь членами порядка $O(\|\delta x\|^2)$. Ищется поправка $\delta x^{1/2}$ к приближению x^0 так, чтобы получить

$$f(x^0 + \delta x^{1/2}) = 0. \quad (2)$$

Разлагая (2) в ряд по δx и отбрасывая величины $O(\|\delta x\|^2)$, получим для $\delta x^{1/2}$ линейную систему уравнений

$$f(x^0) + f_x(x^0)\delta x = 0, \quad (3)$$

откуда

$$\delta x^{1/2} = -f_x^{-1}(x^0)f(x^0), \quad x^1 = x^0 + \delta x^{1/2}. \quad (4)$$

Далее процесс повторяется до получения необходимой точности.

Т е о р е м а 1. Пусть в окрестности искомого решения x^* отображение $f(x)$ равномерно невырождено, т. е.

$$\|f_x^{-1}(x)\| \leq C_1,$$

и пусть также вторые производные $f(x)$ равномерно ограничены в окрестности x^* . Тогда сходимость метода Ньютона в окрестности x^* имеет квадратичный характер.

Точный смысл этого утверждения выяснится в процессе доказательства.

Пусть точное отклонение x^0 от x^* есть малая величина $\Delta x = x^* - x^0$: для нее имеем уравнение

$$0 = f(x^0 + \Delta x) = f(x^0) + f_x(x^0)\Delta x + O(\|\Delta x\|^2).$$

Таким образом,

$$\begin{aligned} x^* &= x^0 - f_x^{-1}(x^0)f(x^0) + f_x^{-1}(x^0)O(\|\Delta x\|^2), \\ x^1 &= x^0 - f_x^{-1}(x^0)f(x^0). \end{aligned}$$

Отсюда

$$x^* - x^1 = f_x^{-1}(x^0)O(\|\Delta x\|^2).$$

В силу ограниченности вторых производных f ,

$$\|O(\|\Delta x\|^2)\| \leq C_2\|\Delta x\|^2$$

и

$$\|x^* - x^1\| \leq C_1C_2\|x^* - x^0\|^2 = C\|x^* - x^0\|^2.$$

Далее,

$$\begin{aligned} \|x^* - x^2\| &\leq C\|x^* - x^1\|^2 \leq C^3\|x^* - x^0\|^4 \\ \|x^* - x^k\| &\leq C^{2^{k-1}}\|x^* - x^0\|^{2^k}. \end{aligned} \quad (5)$$

Именно формула (5) и имеется в виду, когда говорится о квадратичной сходимости. Сходимость гарантируется в окрестности решения $\|x^* - x^0\| \leq C^{-1}$.

Модифицированный метод Ньютона как метод поиска решения. Изложенная выше схема вычислений ориентирована на получение из сравнительно хорошего начального приближения x^0 очень точного значения корня x^* . Однако при грубых начальных данных неоднократно отмечалась расходимость итераций; были предложены усовершенствования, имеющие целью ослабить требования к начальному приближению, расширить окрестность решения x^* , в которой метод сходится.

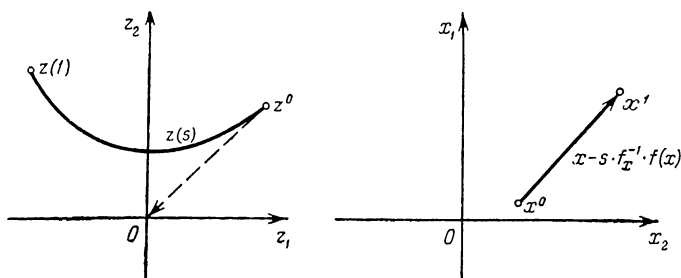


Рис. 69.

Рассмотрим геометрическую интерпретацию метода, взяв для наглядности двумерную задачу ($x = \{x_1, x_2\}$, $f = \{f^1, f^2\}$). Рассмотрим отображение $z = f(x)$ и точку $z^0 = f(x^0)$.

Линеаризуя отображение в окрестности точки z^0 , т. е. заменяя его линейным

$$\tilde{z}(x) = z^0 + f_x(x^0)(x - x^0),$$

находим точку x^1 , такую, что x^1 отображается в $\tilde{z} = 0$. На больших расстояниях $\|x^1 - x^0\| \gg 0$ пренебрегать нелинейностью отображения опасно; например, квадратичные члены могут стать преобладающими и сходимость не будет иметь места. Однако, можно считать $x^1 - x^0 = f_x^{-1}(x^0) z^0$ направлением движения точки x , не предпреляя пока величины смещения x . Другими словами, рассмотрим непрерывное движение от точки x^0 к точке x^1 :

$$x(s) = x^0 - s f_x^{-1}(x^0) z^0, \quad s \geq 0.$$

В плоскости z ему соответствует непрерывное движение точки $z(s) = f[x^0 - s f_x^{-1}(x^0) z^0]$; это движение при $s \simeq 0$ происходит в направлении к точке $z = 0$, как изображено на рис. 69. Постепенно истинное движение $z(s)$ начинает отклоняться от предсказанного на основании линеаризации прямолинейного движения к $z = 0$. Теперь естественно возникает рекомендация — двигаться по отрезку

(x^0, x^1) лишь до тех пор, пока $\|z(s)\|$ убывает, т. е. шаг процесса s^* выбрать, решая, например одномерную задачу минимизации $\|f[x - s f_x^{-1} f(x^0)]\|$. Затем в новой точке $x^1 = x^0 - s^* f_x^{-1} f(x^0)$ находится новое направление движения, и т. д.

Т е о р е м а 2. Пусть в области $\|f(x)\| \leq R$, которую предполагаем ограниченной, $f(x)$ имеет равномерно ограниченное первые и вторые производные; пусть в этой области отображение $z = f(x)$ взаимно однозначно, причем $\|f_x^{-1}(x)\| \leq C$; и пусть в этой области имеется (единственная в силу предыдущих предположений) точка x^* ; $f(x^*) = 0$. Тогда модифицированный метод Ньютона сходится, т. е. $x^k \rightarrow x^*$ при $k \rightarrow \infty$, если $\|f(x^0)\| \leq R$.

Д о к а з а т е л ь с т в о. Рассмотрим последовательность полученных описанной выше процедурой точек $x^0, x^1, \dots, x^k, \dots$. Пусть сходимости x^k к x^* нет. Тогда из $\{x^k\}$ можно выделить сходящуюся к некоторой точке $\tilde{x} \neq x^*$ подпоследовательность. Рассмотрим шаг процесса, начинающийся в точке \tilde{x} : $\tilde{x} \rightarrow \tilde{x}^1$, причем в силу ограниченности вторых производных и невырожденности отображения при малых s ,

$$\begin{aligned} \|f(\tilde{x} - s f_x^{-1} f(\tilde{x}))\| &= \|f(x) - s f_x(\tilde{x}) f_x^{-1}(\tilde{x}) f(\tilde{x}) + O(s^2)\| = \\ &= \|f(\tilde{x}) - s f(\tilde{x}) + O(s^2)\| = (1-s)\|f(\tilde{x})\| + O(s^2). \end{aligned}$$

Таким образом, существует малое число $\varepsilon > 0$ и $\|f(\tilde{x}^1)\| \leq \|f(\tilde{x})\| - \varepsilon$. В силу непрерывности шаг процесса, начинающийся в точке x из некоторой η -окрестности точки \tilde{x} : $\|x - \tilde{x}\| \leq \eta$, сопровождается падением нормы $\|f(x)\|$ не менее, чем на $\varepsilon/2$. Поскольку для точек x^k имеем $\|f(x^0)\| > \|f(x^1)\| > \dots > \|f(x^k)\| > \dots$, а в η -окрестность точки \tilde{x} попадает бесконечное число точек из числа x^k , получаем противоречие с очевидным соотношением $\|f(x)\| \geq 0$.

Таким образом, в принятых предположениях единственной точкой сгущения точек x^k может быть лишь точка x^* , в которой $f(x^*) = 0$. Теорема доказана.

Заметим, что иногда шаг процесса s определяется не решением одномерной задачи минимизации $\|z(s)\|$, а методом деления шага: сначала пробуются точка $x^1 = x^0 - f_x^{-1} f(x^0)$ ($s = 1$). Если $\|f(x^1)\| > \|f(x^0)\|$, то шаг делится пополам и $x^1 = x^0 - \frac{1}{2} f_x^{-1} f(x^0)$; если снова $\|f(x^1)\| \geq \|f(x^0)\|$, рассматривается точка $x^1 = x^0 - \frac{1}{4} f_x^{-1} f(x^0)$ и т. д., до тех пор, пока не будет впервые получена точка $x^1 = x^0 - 2^{-r} f_x^{-1} f(x^0)$, для которой $\|f(x^1)\| < \|f(x^0)\|$.

Используя эту достаточно убедительную расчетную схему при решении, например, краевых задач для нелинейных обыкновенных уравнений, задач линейного программирования и т. д.,

автор неоднократно сталкивался с ситуациями, в которых сходимость была безнадежно медленной. Разбираясь в причинах этого, естественно прежде всего проверить, выполняются ли в точке, в которой застревал процесс поиска, предположения теоремы 2. Основное предположение — это невырожденность отображения $f(x)$, т. е. $\det(f_x(x)) \neq 0$. В ситуациях, о которых идет речь, вырожденности не было, но шаг s процесса был чрезвычайно малым.

Рассмотрим характерный методический пример. Решалась система уравнений

$$f(x, y) \equiv x^5 + y^4 - 2 = 0,$$

$$\varphi(x, y) \equiv (x - 2)^3 + (y - 2)^3 + 16 = 0.$$

Процесс поиска начинался из точки $x^0 = 2$, $y^0 = 3$; ход его отображен в табл. 1 следующими величинами: номер шага k , полученные на k -м шаге значения x , y , f , φ , $F = \sqrt{(f^2 + \varphi^2)}$, шаг s^* при переходе от k -й точки к $(k + 1)$ -й и α — угол (в градусах)

Т а б л и ц а 1

k	x	y	f	φ	F	s	α
0	2,0000	3,0000	111,000	17,000	112,294	0,11	40
1	2,0685	2,9380	110,379	16,826	111,654	0,108	42
2	2,1330	2,8680	109,801	16,656	111,057	0,094	48
3	2,1872	2,7976	109,305	16,515	110,545	0,075	53
4	2,2334	2,7274	108,906	16,398	110,134	0,064	57
5	2,2727	2,6589	108,607	16,306	109,824	0,047	52
6	2,3051	2,5946	108,395	16,2386	109,605	0,027	49
7	2,3265	2,5471	108,246	16,199	109,451	0,0205	46
8	2,3437	2,5057	108,138	16,170	109,340	0,0150	43
10	2,3729	2,4284	108,008	16,130	109,206	0,012	33
12	2,3899	2,3781	107,946	16,113	109,141	0,0040	25
14	2,4031	2,3358	107,914	16,103	109,109	0,0017	18
16	2,4108	2,3096	107,898	16,099	109,092	0,0010	15

k	x	y	f	φ	F_α	F_ω	s
0	2,0000	3,0000	111,000	17,000	5,72	5,52	0,066
1	2,414	2,626	127,500	16,32	12,75	11,59	0,125
2	3,209	0,542	338,4	14,66	2,00	0,55	0,50
3	2,891	-0,389	200,0	3,07	0,600	0,0826	1,90
4	1,806	-0,578	17,3	-1,14	0,331	0,026	2,00
5	1,157	-0,460	0,115	0,516	0,031	0,00054	1,0
6	1,14265	-0,48663	0,004	-0,006	0,0006	0	1,0
7	1,14220	-0,48626	0,00000	0,00000			

между векторами $\{f_x, f_y\}$ и $\{\varphi_x, \varphi_y\}$ (он характеризует невырожденность системы). В точке x^0, y^0 система уравнений для направления спуска имела вид

$$\begin{pmatrix} f_x & f_y \\ \varphi_x & \varphi_y \end{pmatrix} \begin{Bmatrix} \delta x \\ \delta y \end{Bmatrix} = - \begin{Bmatrix} f \\ \varphi \end{Bmatrix} : \begin{pmatrix} 80 & 108 \\ 0 & 3 \end{pmatrix} \begin{Bmatrix} \delta x \\ \delta y \end{Bmatrix} = - \begin{Bmatrix} 117 \\ -17 \end{Bmatrix}.$$

Характерным является соотношение

$$f_x^2 + f_y^2 \gg \varphi_x^2 + \varphi_y^2, \quad (6)$$

выполнявшееся и в других точках $\{x^k, y^k\}$, представленных в таблице. Решение этой системы $\delta x = 6,26$, $\delta y = -5,67$ почти ортогонально вектору $\{f_x, f_y\}$: $f_x \delta x + f_y \delta y \approx 500 - 611 = -111$; таким образом, направление спуска $\{\delta x, \delta y\}$ близко к направлению линии уровня функции $f(x, y)$, и вдоль такого направления первоначальное падение $f(x, y)$ при малых s сменяется ростом; хотя при этом продолжается падение функции $\varphi(x, y)$, в целом норма $F = \sqrt{f^2 + \varphi^2}$ начинает возрастать, так как в силу (6) вклад приращения f в приращение F выражается существенно большим числом, чем вклад приращения φ .

Заметим, что как сама задача, так и направление спуска $\{\delta x, \delta y\}$ инвариантны относительно простейшего преобразования — изменения единиц измерения f и φ :

$$\begin{aligned} f(x, y) &\rightarrow \mu_1 f(x, y), \\ \varphi(x, y) &\rightarrow \mu_2 \varphi(x, y). \end{aligned} \quad (7)$$

Однако норма $F = (f^2 + \varphi^2)^{1/2}$ и, следовательно, шаг процесса s , относительно преобразования (7) не инвариантны, и возникает вопрос о разумной нормировке задачи. В данном случае, как и во многих других аналогичных ситуациях, автор руководствовался следующим естественным соображением: нужно нормировать задачу так, чтобы порожденные вариациями аргументов $\delta x, \delta y$ вариации функций

$$\delta f = f_x \delta x + f_y \delta y, \quad \delta \varphi = \varphi_x \delta x + \varphi_y \delta y$$

были величинами одного порядка. Теперь понятно, почему в качестве нормы, минимизация которой вдоль направления спуска определяла шаг процесса, бралась величина

$$F \equiv \left(\frac{1}{\mu_1} f^2 + \frac{1}{\mu_2} \varphi^2 \right), \quad (8)$$

где $\mu_1 = f_x^2 + f_y^2$, $\mu_2 = \varphi_x^2 + \varphi_y^2$.

Результат этой нормировки не замедлил сказаться; в табл. 1 приведен ход решения задачи.

Заметим, что теперь в задаче нет единой нормы вектора $\{f, \varphi\}$, которая монотонно убывает и этим обеспечивается сходимость процесса. Едва ли удастся доказать сходимость в предположениях теоремы 2, если шаг s определяется минимизацией «локальной» нормы (8). Однако как в этой, так и во многих других задачах, нормировка типа (8) оказывалась чрезвычайно полезной и помогала (часто решающим образом) преодолеть медленную сходимость. В то же время случаев, когда такая нормировка приводит к расходимости, не встречалось.

В табл. 1 приведены значения F_α в исходной точке шага, и F_ω — в конечной; F_ω — есть минимум F на направлении спуска. Характерным является резкий рост f , существенно превышающий падение φ вдоль направления спуска. Однако теперь мы на рост f не обращаем (до известной степени) внимания: соотношение (6) свидетельствует о том, что f очень легко изменить сравнительно малыми вариациями δx , δy ; главное — это вывести на нуль φ . Эти качественные соображения и учитывает нормировка (8). Следует заметить, что нормировка (8) не является безусловно обязательной. Та же система при других начальных данных $\{x^0, y^0\}$ одинаково хорошо решалась как с нормировкой, так и без нее. В значительной мере это связано с тем, что в большинстве точек $\{x, y\}$ система уже нормирована. Однако в методических целях мы можем ее «испортить», заменив уравнения на

$$10 \cdot f(x, y) = 0; \quad 0,1 \cdot \varphi(x, y) = 0. \quad (9)$$

Решение этой задачи представлено в табл. 2 теми же величинами; кроме того, добавлено n — число вычислений функций f и φ , понадобившиеся для выбора шага s . Во второй части табл. 2 представлено решение той же задачи с использованием нормировки (8). Видно, что нормировка оказалась полезной, хотя и без нее процесс сошелся. Разумеется, в такой простой задаче можно заранее отклонить формулировку (9) задачи как неестественную. Однако в сложных задачах, когда f и φ определяются не легко обозримыми формулами, а сложными вычислительными процессами, и являются величинами разных, например, физических размерностей, заранее не ясно, является ли содержательный выбор единиц измерения f и φ естественным и с вычислительной точки зрения.

З а м е ч а н и е о д и ф ф е р е н ц и а л ь н ы х у р а в н е н и я х с п у с к а. Иногда процесс поиска решения оформляется в виде дифференциальных уравнений движения точки $x(s)$:

$$\frac{dx}{ds} = -f_x^{-1}(x)f(x), \quad x(0) = x^0, \quad (10)$$

Т а б л и ц а 2

h	x	y	f	φ	F	s	n
0	1,0000	1,3000	18,56	1,466	18,619	0,0016	6
1	0,98939	1,30570	18,546	1,463	18,603	0,0016	6
2	0,97936	1,31076	18,528	1,461	18,586	0,0016	6
3	0,96980	1,31531	18,509	1,459	18,567	0,0032	7
4	0,95150	1,32355	18,487	1,454	18,544	0,0032	7
5	0,93455	1,33041	18,457	1,449	18,514	0,008	5
6	0,89476	1,34495	18,456	1,437	18,512	0,008	5
7	0,85990	1,35492	18,403	1,425	18,458	0,008	5
8	0,82836	1,36210	18,322	1,413	18,377	0,016	6
9	0,77031	1,37272	18,220	1,389	18,273	0,04	4
10	0,64404	1,38716	18,134	1,328	18,183	0,20	3
11	0,16164	1,39206	17,553	0,956	17,579	1,0	7
12	-0,76376	1,22968	0,266	-0,557	0,617	0,4	4
13	-0,66470	1,20559	-0,1726	-0,342	0,383	0,4	4
14	-0,59972	1,19752	-0,210	-0,209	0,296	0,8	6
15	-0,51686	1,19217	-0,169	-0,047	0,175	1,0	7
16	-0,49223	1,19336	-0,0079	-0,00046	0,008	1,0	7
17	-0,49199	1,19347	0,00000	0,00000	0,00000		

h	x	y	f	φ	F	s	n
0	1,00000	1,30000	18,56	1,466	4,391	0,2	3
1	-0,32615	2,01228	143,93	0,341	0,489	1,2	8
2	-0,57846	1,48281	27,696	-0,128	0,222	1,4	9
3	-0,47640	1,18110	-0,785	0,0264	0,0186	1,0	7
4	-0,49213	1,19363	0,0105	-0,00022	0,00019	1,0	7
5	-0,49199	1,19347	0,00000	0,00000	0,00000		

и доказывается (в предположениях, аналогичных предположениям теоремы 2), что решение системы, — точка x^* , — является единственной асимптотически устойчивой точкой системы дифференциальных уравнений (10). Это является очевидным следствием известной теоремы Ляпунова; в качестве функции Ляпунова берется $V(x) \equiv \|f(x)\|^2$. Повторяя дословно выкладки из доказательства теоремы 2, получим

$$\frac{dV[x(s)]}{ds} = -V[x(s)], \text{ т. е. } V[x(s)] = V[x^0] e^{-s}. \quad (11)$$

Иногда этот результат трактуют как некоторую оценку скорости поиска решения. Однако никакой полезной с этой точки зрения информации формула (11) не содержит; она характеризует лишь способ введения параметра s на траектории:

$$dx_1 : dx_2 : \dots : dx_n = F_1 : F_2 : \dots : F_n, \quad F(x) = -f_x^{-1}(x)f(x).$$

Если всерьез полагать, что формула (11) характеризует эффективность процесса поиска решения, то следовало бы еще больше «повысить» ее, заменив (10), например, уравнением $\dot{x} = -10^6 \times \times f_x^{-1} f$, т. е. $V[x(s)] = V_0 e^{-10^6 \cdot s}$. Ясно, что эффективность поиска связана не с характером параметризации траектории (12), а с затратами реального машинного времени, т. е. с числом вычислений функций $f(x)$, $f_x(x)$, затрачиваемых на достижение достаточно малых значений $\|f(x)\|$. С практической точки зрения теорема 2 гарантирует успех процесса поиска с достаточно малым шагом s ; в этом случае процесс типа

$$x^{k+1} = x^k - s f_x^{-1}(x^k) f(x^k) \quad (12)$$

есть интегрирование (10) методом Эйлера. Но достаточно малый шаг s , необходимый для правильного описания ломаной (12) решения уравнения (10), нам совершенно не нужен: нас интересует лишь убывание $\|f(x)\|$. Основное машинное время в задаче связано с вычислением матрицы $f_x(x)$, позволяющей найти направление спуска, и естественно стремление получить от этого направления все, что оно может дать, т. е. выбрать шаг s минимизацией $\|f(x + s\delta x)\|$, а не в интересах точности интегрирования уравнений спуска. Есть и полезный аспект перехода к уравнениям (10) — это использование методов интегрирования высокого порядка точности, позволяющих при достаточной гладкости $f(x)$ использовать значительно больший шаг, чем в процессе первого порядка точности (12). Однако и здесь, видимо, следует отделить полезное с точки зрения интересующего нас здесь результата — минимизации $\|f(x)\|$, от ненужной точности в самой траектории $x(s)$. Сделать это можно, например, так. Известно, что основу метода Рунге—Кутты составляет разложение траектории $x(s)$ в ряд в окрестности очередного счетного узла. Аналогичными вычислениями можно найти в окрестности очередной точки x^k процесса поиска разложение траектории спуска $x(s)$:

$$x(s) = x^k + s a_1(x^k) + s^2 a_2(x^k) + \dots + s^r a_r(x^k) + O(s^{r+1}),$$

и поиск шага s осуществить задачей одномерной минимизации

$$\min_s \|f(x^k + a_1 s + a_2 s^2 + \dots + a_r s^r)\|.$$

Автору неизвестно, была ли кем-нибудь реализована подобная схема вычислений.

З а м е ч а н и е о б у н и в е р с а л ь н о й п о с л е д о в а т е л ь н о с т и ш а г о в. В чисто теоретических исследованиях большой популярностью пользуется конструкция последовательных шагов спуска $s^{k+1/2}$, определяющих переход от x^k к x^{k+1} :

$$x^{k+1} = x^k - s^{k+1/2} f_x^{-1}(x^k) f(x^k), \quad (13)$$

избавляющая (при доказательстве теорем!) от всяких хлопот с выбором шага. Именно, шаги $s^{k+1/2}$ должны образовывать сходящийся ряд, члены которого стремятся к нулю. Например, $s^{k+1/2} = \frac{1}{k+1}$; эта конструкция содержит две полезные идеи, почти очевидные. Во-первых, шаги спуска должны убывать и стремиться к нулю, иначе процесс типа (13) будет проскакивать точку минимума x^* и закончится «болтанкой» в ее окрестности: размер окрестности зависит от шага s . Однако $s^{k+1/2}$ не должны убывать слишком быстро: если ряд $\sum_{k=0}^{\infty} s^{k+1/2}$ сходится, то пройденного точкой x^k , $k=0, 1, \dots$ за бесконечное число шагов (т. е. за бесконечное машинное время) пути может просто не хватить для перехода из x^0 в x^* .

Однако практическая ценность таких универсальных последовательностей шагов — незначительна. В самом деле, сохраняя указанные свойства последовательности $s^{k+1/2}$, можно умножить или разделить все $s^{k+1/2}$ на 10^6 , например, добавить впереди миллион единиц, выбросить в любом месте миллион очередных членов и т. д. Все эти преобразования, совершенно не существенные с чисто теоретической точки зрения, оказывают решающее влияние на эффективность процесса поиска. Ясно, что выбор последовательности шагов $s^{k+1/2}$ должен определяться не априорными универсальными конструкциями, а достаточно квалифицированными алгоритмами обратной связи, использующими вычисляемые в процессе поиска значения $f(x)$.

§ 44. Дискретное динамическое программирование

Основным содержанием настоящего параграфа является алгоритм динамического программирования, позволяющий эффективно решать специальные дискретные задачи оптимального управления. Такие задачи могут появляться при оптимизации дискретных систем и при аппроксимации задач оптимального управления (см., например, § 15).

Д и с к р е т н а я з а д а ч а у п р а в л е н и я. Рассматривается последовательность моментов времени $0, 1, \dots, N$. Управляемая система в каждый момент времени n может находиться в одном из J состояний. Управление системой, находящейся в момент времени n в состоянии j_n , состоит в том, что принимается решение о переводе ее в момент $n+1$ в состояние j_{n+1} . Определена локальная цена такого перехода — число $f_{j_n, j_{n+1}}^{n+1/2}$ (для всех возможных пар j_n, j_{n+1}). В начальный момент времени $n=0$ система может находиться в любом из J_0 состояний (или в каком-то фиксированном: в этом случае можно считать множество состояний

J_0 состоящим из одной точки). В конечный момент времени N система должна находиться в одном из заданных J_N состояний. Задача состоит в определении такой последовательности состояний

$$j_0, j_1, \dots, j_n, \dots, j_N \quad (1)$$

(эту последовательность будем называть *траекторией*), которая минимизирует общую цену эволюции системы, т. е. функцию

$$R(j_0, j_1, \dots, j_N) \equiv \Phi_0(j_0) + \sum_{n=0}^{N-1} f_{j_n, j_{n+1}}^{n+1/2} + \Phi_N(j_N), \quad (2)$$

где Φ_0, Φ_N — «плата» за стартовое и финишное состояние системы.

Решение задачи осуществляется специальным алгоритмом, использующим типичную для динамического программирования функцию Беллмана $F_n(j)$. Определяется она следующим образом. Рассмотрим часть задачи: пусть система в момент n находится в состоянии j . Нужно перевести ее к моменту N , минимизируя за счет выбора состояний j_{n+1}, \dots, j_N значение

$$\sum_{k=n}^{N-1} f_{j_k, j_{k+1}}^{k+1/2} + \Phi_N(j_N), \quad j_n = j. \quad (3)$$

Минимум (3) и обозначается $F_n(j)$. Оказывается (и это тоже характерный для динамического программирования факт), что $F_n(j)$ удовлетворяет уравнению динамического программирования, используя которое можно вычислить функции $F_n(j)$ для всех n и j . Это уравнение выводится на основании «принципа оптимальности»: переход из состояния j в момент n в какое-то состояние j_N в момент N можно осуществить в два этапа: сначала система переводится в состояние i в момент $n+1$, а затем из этого состояния оптимальным образом за цену $F_{n+1}(i)$ — в конечное состояние. Общая стоимость такого перехода есть, очевидно,

$$f_{j,i}^{n+1/2} + F_{n+1}(i), \quad (4)$$

и в (4), так как j мы считаем фиксированным, есть лишь один параметр оптимизации — номер i состояния в момент $n+1$. Тогда

$$F_n(j) = \min_i \{f_{j,i}^{n+1/2} + F_{n+1}(i)\}, \quad (5)$$

Это и есть уравнение динамического программирования.

Алгоритм решения задачи (5) представляет собой процедуру последовательного вычисления функций.

1. Функция $F_N(j)$ задается «начальными данными»:

$$F_N(j) = \Phi_N(j).$$

2. Для определения $F_{N-1}(j)$ используем (5):

$$F_{N-1}(j) = \min_i \{f_{j,i}^{N-1/2} + F_N(i)\}. \quad (6)$$

Если решать эту задачу самым простым способом — перебором, то вычисление $F_{N-1}(j)$ обойдется в $O(J^2)$ операций, а описание функции $F_{N-1}(j)$ потребует J ячеек памяти. Далее продолжаем в том же духе, и, получив функцию $F_{n+1}(j)$, вычисляем:

$$F_n(j) = \min_i \{f_{j,i}^{n+1/2} + F_{n+1}(i)\}. \quad (7)$$

Кстати, здесь же определяются и функции управления $i_{n+1/2}(j)$, указывающие, в какое состояние i следует в момент $n+1$ перевести систему, если в момент n она окажется в состоянии j . Найдя таким образом $F_0(j)$, следует решить задачу $\min_j \{\Phi_0(j) + F_0(j)\}$ и определить первую точку траектории j_0 . Затем определяются точки $j_1 = i_{1/2}(j_0)$, $j_2 = i_{1+1/2}(j_1)$ и т. д. Стоимость решения в общем случае есть $O(N^2)$ операций, реализация требует NJ ячеек памяти (или $2NJ$, если запоминать и функцию $i_{n+1/2}(j)$).

Непрерывная задача динамического программирования. В принципе, алгоритм динамического программирования применим и в том случае, когда состояние системы в момент времени n (остающийся здесь дискретным) описывается точкой x_n r -мерного пространства, положение которой ограничено условием $x_n \in G_n$. В этом случае, точно так же, как было получено уравнение (5), можно получить и его непрерывный вариант:

$$F_n(x_n) = \min_{x_{n+1} \in G_{n+1}} \{f^{n+1/2}(x_n, x_{n+1}) + F_{n+1}(x_{n+1})\}. \quad (8)$$

Однако фактическая реализация этого алгоритма наталкивается на серьезное препятствие: нет никаких оснований ожидать, что все функции F_n будут получены в замкнутой аналитической форме, попытки же заменить функциональные зависимости таблицами (что приводит к дискретному варианту задачи) наталкиваются на препятствие, метко названное Р. Беллманом «проклятием размерности»: при $r > 2$ задача практически становится непосильной для современных ЭВМ. Однако есть частный случай, когда уравнение (8) тем не менее решается в конечном виде. Это задачи, в которых все функции $f^{n+1/2}(x_n, x_{n+1})$ квадратичны и, если начальное и конечное состояния не фиксированы, хотя бы одна из функций (пусть $\Phi_N(x_N)$) — квадратичная. Кроме того, области G являются полным r -мерным пространством. Этот случай, несмотря на определенную искусственность, может оказаться полезным

при построении аппроксимационных процедур решения более общей задачи. Введем обозначения (при $r=1$)

$$\Phi_N(x_N) = \frac{1}{2} Ax_N^2 + Bx_N + C, \quad (9)$$

$$f^{n+1/2}(x_n, x_{n+1}) = ax_n^2 + 2bx_nx_{n+1} + cx_{n+1}^2 + \alpha x_n + \beta x_{n+1} + \gamma. \quad (10)$$

Ради простоты мы опустили в (10) индекс $n + 1/2$ при коэффициентах a, b, \dots, γ .

Теорема 1. В непрерывной задаче динамического программирования с функциями (9), (10) все функции $F_n(x_n)$ — суть квадратичные формы:

$$F_n(x_n) = r_n x_n^2 + p_n x_n + q_n. \quad (11)$$

Коэффициенты r_n, p_n, q_n вычисляются по простым рекуррентным формулам (12), (13).

Доказательство. Пусть $F_n(x_n)$ имеет форму (11) и коэффициенты r_n, p_n, q_n уже найдены. Найдем F_{n-1} , решив уравнение

$$\begin{aligned} F_{n-1}(x) &= \min_y \{f^{n-1/2}(x, y) + F_n(y)\} = \\ &= \min_y \{ax^2 + 2bxy + cy^2 + \alpha x + \beta y + \gamma + r_n y^2 + p_n y + q_n\}. \end{aligned}$$

Очевидным образом получим точку минимума y в виде линейной формы от x :

$$y = l_n x + m_n, \quad \text{где } l_n = \frac{b_{n-1/2}}{r_n + c_{n-1/2}}, \quad m_n = -\frac{\beta_{n-1/2} + p_n}{2(r_n + c_{n-1/2})}. \quad (12)$$

Подставляя (12) в уравнение динамического программирования, получим $F_n(x)$ в форме (11) с коэффициентами

$$\begin{aligned} r_{n-1} &= a_{n-1/2} + 2b_{n-1/2}l_n + cl_n^2 + r_n l_n^2, \\ p_{n-1} &= 2bm_n + 2cl_n m_n + \alpha + \beta l_n + 2r_n l_n m_n + p_n l_n, \\ q_{n-1} &= cm_n^2 + \beta m_n + r_n m_n^2 + p_n m_n + \gamma q_n \end{aligned} \quad (13)$$

(все коэффициенты a, b, c, \dots, γ в формулах (12), (13) должны быть снабжены индексом $n - 1/2$, который мы ради простоты опустили). При произвольном r исследование аналогично.

§ 45. Поиск минимума. Гладкие задачи

Здесь будут рассмотрены основные идеи, которые используются при численном решении следующих типичных задач оптимизации. Они перечислены в порядке возрастания формальной сложности.

Задача 1. Задана функция $f^0(x)$ векторного аргумента $= \{x_1, x_2, \dots, x_n\}$. Найти точку x^* минимума $f^0(x)$:

$$f^0(x^*) \leq f^0(x) \text{ при всех } x. \quad (1)$$

В этой задаче на значения переменных x никаких ограничений не наложено, поэтому о ней говорят, как о задаче безусловной минимизации.

Задача 2. Найти минимум $f^0(x)$ в некоторой части n -мерного пространства X (X предполагается, естественно, замкнутым множеством):

$$\min_{x \in X} f^0(x). \quad (2)$$

Задача 3. Найти минимум $f^0(x)$ в области X ; кроме того, должны быть выполнены дополнительные условия $f^i(x)=0$, $i=1, 2, \dots, m$;

$$\min_{x \in X} f^0(x) \text{ при условиях } f^i(x)=0, \quad i=1, 2, \dots, m. \quad (3)$$

Эту задачу называют задачей условной минимизации $f^0(x)$. В качестве отдельной задачи мы выделим случай, когда некоторые из дополнительных условий имеют форму неравенств.

Задача 4. Найти

$$\min_{x \in X} f^0(x) \text{ при условиях } f^i(x)=0 (\leq 0). \quad (4)$$

Мы будем считать все функции f^i , ($i=0, 1, \dots, n$), входящие в постановку задачи, достаточно гладкими, т. е. имеющими непрерывные производные до того порядка, который будет необходим при тех или иных выкладках. Этот случай мы будем считать стандартным и не требующим специальных оговорок. Наоборот, если та или иная из нужных нам производных может не существовать в отдельных точках x , будем считать, что в этом случае требуется специальное предупреждение.

Стоит еще отметить, что различие между задачами (2), (3) и (4) кажется условным. Ведь всегда можно определить множество X как совокупность точек, удовлетворяющих условиям $f^i(x)=0$ (≤ 0). С другой стороны, фактически область X определяется именно набором подобных равенств и неравенств. Тем не менее в специальном выделении множества X имеется определенный смысл, если геометрия области X очень проста. Критерием простоты является простота важной в дальнейшем операции *проектирования* на X .

Определение 1. Пусть X — замкнутое множество точек n -мерного пространства, x — произвольная точка этого прост-

ранства. Точку y будем называть *проекцией* x на X , если она является ближайшей к x точкой из X :

$$\|y - x\| = \min_{x' \in X} \|x' - x\|. \quad (5)$$

Для операции проектирования на X будем использовать обозначение P_X ; таким образом, $y = P_X(x)$. Разумеется, операция проектирования зависит от используемой нормы. Если не оговорено противное, мы имеем в виду евклидову норму. Очень часто «простая» область X определяется заданием для каждой компоненты x двусторонних ограничений

$$\{x \in X \text{ эквивалентно } x^- \leq x \leq x^+\}. \quad (6)$$

В этом случае операция проектирования на X приводит к задаче

$$\min_{x^- \leq y \leq x^+} \sum_{i=1}^n (y_i - x_i)^2.$$

Решение очевидно (оно получается операцией «срезки» x):

$$y_i = \begin{cases} x_i^+, & \text{если } x_i > x_i^+, \\ x_i^-, & \text{если } x_i < x_i^-, \\ x_i, & \text{если } x_i^- \leq x_i \leq x_i^+, \end{cases} \quad i = 1, \dots, n. \quad (7)$$

Простой является и сфера

$$\left\{x \in X \text{ эквивалентно } \sum_{i=1}^n x_i^2 \leq 1\right\}. \quad (8)$$

В этом случае проектирование приводит к задаче

$$\min_y \sum_{i=1}^n (y_i - x_i)^2 \text{ при условии } \sum_{i=1}^n y_i^2 \leq 1. \quad (9)$$

Образуя функцию Лагранжа (см. ниже)

$$\mathcal{L}(y, \lambda) = \sum_{i=1}^n [(y_i - x_i)^2 + \lambda y_i^2],$$

получаем выражения для y_i :

$$\frac{\partial \mathcal{L}}{\partial y} = 0 = y_i - x_i + \lambda y_i, \text{ т. е. } y_i = \frac{1}{1 + \lambda} x_i,$$

а параметр λ находится из условия

$$\frac{1}{(1 + \lambda)^2} \sum_{i=1}^n x_i^2 = 1, \text{ т. е. } 1 + \lambda = 1/\|x\|$$

(второе решение $1 + \lambda = -1/\|x\|$ дает максимум $\|y - x\|$). Несколько более сложным является случай, когда область задается общей квадратичной формой:

$$\{x \in X \text{ эквивалентно } \frac{1}{2}(Ax, x) + (b, x) \leq 1\}. \quad (10)$$

В этом случае проектирование связано с задачей

$$\min_y \sum_{i=1}^n (y_i - x_i)^2 \text{ при условии } \frac{1}{2}(Ay, y) + (b, y) \leq 1. \quad (11)$$

Функция Лагранжа

$$\mathcal{L}(g, \lambda) \equiv \frac{1}{2} \sum_{i=1}^n (y_i - x_i)^2 + \lambda \left(\frac{1}{2}(Ay, y) + (b, y) \right)$$

дает нам следующие уравнения:

$$\frac{\partial \mathcal{L}}{\partial y} = y - x + \lambda \frac{A + A^*}{2} y + \lambda b = 0. \quad (12)$$

Это есть система линейных алгебраических уравнений относительно y_i ; коэффициенты системы содержат неизвестный параметр λ , значение которого определяется условием $y \in X$ (11). Имея в виду общий случай, мы можем лишь при разных значениях λ решать систему и затем подбирать значение λ так, чтобы было выполнено условие (11). Речь идет о решении скалярного равенства $R(\lambda) = 1$, где функция $R(\lambda)$ определена следующим алгоритмом вычисления.

1. При заданном λ решается система (12), находятся значения $y(\lambda)$.

2. Вычисляется $R(\lambda) = \frac{1}{2}(Ay, y) + (b, y)$ (или, что то же самое в силу (12): $R(\lambda) = \frac{1}{\lambda}(x - y, y)$). Далее, если $x \notin X$, что легко проверяется, следует решить уравнение $R(\lambda) = 1$, используя, например, метод Ньютона.

Вычислительная сложность этой задачи проектирования (11) определяется в основном размерностью пространства n : если n достаточно велико, задача может оказаться очень сложной, так как в общем случае трудоемкость решения системы n линейных уравнений составляет $O(n^3)$ операций.

А л г о р и т м ы б е з у с л о в н о й м и н и м и з а ц и и.
Теперь мы приступим к описанию основных алгоритмов решения задач минимизации. Это алгоритмы спуска, в которых, имея некоторую точку x^0 , находим рядом с ней другую точку x^1 , такую, что $f(x^1) < f(x^0)$. Если это невозможно, у нас есть основания утверждать, что найдена точка минимума $f(x)$.

Нужно подчеркнуть два характерных для всех методов поиска обстоятельства:

1) речь может идти лишь о локальном минимуме функции $f(x)$, поиск глобального минимума существенно труднее;

2) не всегда, когда мы сталкиваемся с ситуацией, в которой метод не находит следующей «лучшей» точки x^1 , можно говорить о локальном минимуме: точка x^0 может быть тупиковой точкой для данного метода построения последовательности x^0, x^1, x^2, \dots

В общем случае мы получаем последовательность точек $x^0, x^1, x^2, \dots, x^k, \dots$ с монотонно убывающими значениями $f(x^0) > f(x^1) > \dots > f(x^k) > \dots$. Эта последовательность называется минимизирующей, если

$$f(x^k) \rightarrow \min_x f(x) \quad \text{при} \quad k \rightarrow \infty.$$

Используя тот или иной метод построения последовательности x^0, x^1, \dots , следует четко представлять себе, какие точки являются тупиковыми для данного метода, не являясь при этом точками локального минимума.

Одномерный поиск минимума. Мы начнем с простой задачи, в которой аргумент x — скаляр. Эта задача полезна постольку, поскольку она появляется в качестве элемента в более сложных задачах.

Алгоритм *параболической аппроксимации* рассчитан на достаточно гладкие функции $f(x)$. Пусть имеется некоторая точка x^0 . Положим $s_1 = x^0$, $s_2 = x^0 + h$, $s_3 = x^0 + 2h$ и вычислим значения $f_i = f(s_i)$, $i = 1, 2, 3$.

Через полученные точки проводится парабола $as^2 + bs + c$, аппроксимирующая $f(s)$, и проверяется ее «выпуклость». Если $a > 0$, то в качестве следующей точки s_4 берется точка минимума этой параболы: $s_4 = -b/2a$, строится новая парабола по точкам $\{s_i, f_i\}$, $i = 2, 3, 4$ и т. д. до стабилизации значения $f_k = f(s_k)$. Если $a < 0$, то следует выяснить направление убывания f , сравнив, например, $f(s_1)$ с $f(s_2)$, и в качестве точки s_4 взять $s_1 + h$ или $s_1 - h$.

Используя этот простой алгоритм, следует решить еще несколько технологических вопросов:

- 1) каким брать начальный шаг h ;
- 2) если расстояние от s_3 до s_4 слишком велико, его следует ограничить;
- 3) если $a < 0$, то продвижение на h может оказаться слишком медленным (при малом h); естественно ввести какие-то алгоритмы подбора h , основанные на анализе фактически реализуемых значений.

Сказанное выше выглядит вполне естественным и убедительным, но читатель, имеющий намерение реализовать эти указания в вычислениях, сразу же столкнется с необходимостью введения

количественных критериев для выражений «если h слишком мало», или «если h слишком велико». Алгоритм параболической аппроксимации неоднократно использовался автором в различных расчетах (см., например, §§ 26, 27, 43), и обычно никаких затруднений не возникало, разумная величина h устанавливалась, в частности, в процессе отладки программы. Заметим, кстати, что величина h определена, в сущности, лишь с точностью до порядка. Пожалуй, единственным объективным критерием для h , используемым автором, является требование к точности аппроксимации. В самом деле, построив полином $f^*(s) = as^2 + bs + c$, аппроксимирующий $f(s)$, и вычисляя в дальнейшем $f(s_i)$, мы можем сравнивать $f(s_i)$ с $f^*(s_i)$. Если ошибка аппроксимации не превосходила, скажем, 10%, ситуация считалась нормальной, и ограничение на шаг h не вводилось. В ситуации $a < 0$ в случае продвижения на h мы также можем, сравнивая $f(s \pm h)$ с $f^*(s \pm h)$, судить о том, не следует ли h увеличить. Если f и f^* совпадают слишком хорошо (например, ошибка $\sim 1\%$ или меньше), h увеличивается.

Методы спуска. Общая схема этих алгоритмов построена следующим образом: Пусть имеется некоторая точка x^k , полученная на k -м шаге процесса поиска минимума.

1. Строится направление спуска y^k .
2. Находится шаг спуска s^k .
3. Точка x^{k+1} вычисляется по формуле

$$x^{k+1} = x^k + s^k y^k.$$

Существует большое число различных рецептов построения направлений спуска и относительно небольшое число способов вычисления шага спуска. Мы ограничимся здесь одним, по существу, способом вычисления s . Будем считать, что s^* находится решением задачи

$$\min_y f(x^k + sy^k). \quad (13)$$

Разумеется, эта задача решается приближенно. Что касается направления спуска, то наиболее популярными являются следующие рецепты.

1. *Метод случайного спуска:* в качестве y берется случайный вектор единичной длины (т. е. случайная точка на поверхности единичной сферы в n -мерном пространстве; обычно для этой точки принимается равномерное распределение).

2. *Метод покоординатного спуска:* векторы y^k образуют n -периодическую последовательность ортов в n -мерном пространстве:

$$e^1 = \{1, 0, \dots, 0\}, e^2 = \{0, 1, 0, \dots, 0\}, \dots, e^n = \{0, \dots, 0, 1\}.$$

3. *Метод наискорейшего спуска*: в качестве y^k берется направление $y^k = -f_x(x^k)$. Это направление получило название направления наискорейшего спуска, так как именно оно находится решением следующей естественной задачи:

$$\min_{\|y\|=\varepsilon} \{f(x) + f_x(x)y\}: \quad y = -\varepsilon \frac{f_x(x)}{\|f_x(x)\|}, \quad (14)$$

являющейся линеаризацией исходной задачи в ε -окрестности точки x .

Прежде всего выясним характер *стационарных точек* этих процессов, т. е. тех точек x , в которых данный процесс «застревает» и $x^{k+1} = x^k$.

1. Стационарной точкой случайного спуска является точка локального минимума $f(x)$, т. е. точка, в которой $f_x(x) = 0$, а матрица вторых производных, если она невырождена, положительно определена.

2. Стационарной точкой покоординатного спуска является точка, в которой $f_x(x) = 0$, и, кроме того, положительны лишь диагональные элементы матрицы f_{xx} (ради простоты мы не анализируем различных вырождений, связанных с обращением в нуль каких-то вторых производных).

3. Стационарной точкой наискорейшего спуска является точка, в которой $f_x(x) = 0$ (т. е. например, любая точка перегиба).

Эти утверждения вполне очевидны, и мы их доказывать не будем. Отметим лишь, что чем уже множество стационарных точек, тем надежнее метод поиска. С этой точки зрения предпочтителен случайный поиск. Однако с точки зрения эффективности предпочтительнее метод наискорейшего спуска. Не исключенная в принципе опасность «застрять» в точке перегиба маловероятна, так как такие точки являются неустойчивыми точками метода наискорейшего спуска, в отличие от точки локального минимума, являющейся устойчивой.

Теперь мы докажем характерную теорему о сходимости метода наискорейшего спуска. Эта теорема, в частности, позволит уточнить требования к точности определения шага спуска s . Обозначим через $R(y)$ область n -мерного пространства, определяемую условием $x \in R(y): f(x) \leq f(y)$. Функцию $f(x)$ будем считать гладкой (достаточно непрерывности вторых производных).

Т е о р е м а. Пусть $R(x^0)$ — ограниченная замкнутая область, и $f(x)$ в $R(x^0)$ имеет лишь единственную точку x^* , в которой $f_x(x^*) = 0$; эта точка, таким образом, является единственной точкой локального минимума $f(x)$ в $R(x^0)$. Тогда метод наискорейшего спуска, стартующий из точки x^0 , сходится к x^* .

Другими словами, метод строит последовательность $x^0, x^1, \dots, x^k, \dots \rightarrow x^*$. Доказательство использует два основных факта.

1. Значения функции f на последовательности $\{x^k\}$ монотонно убывают (если $x^k \neq x^*$):

$$f(x^0) > f(x^1) > \dots > f(x^k) > \dots$$

Это следует из предположения, что $f_x = 0$ лишь в точке x^* .

2. Определим $\Delta(x)$ — функцию выигрыша:

$$\Delta(x) \equiv \min_s f(x - sf_x(x)) - f(x),$$

т. е. величину убывания $f(x)$ при переходе из точки x в точку $x - sf_x(x)$, где s — шаг спуска, вычисленный в соответствии с (13). Несложно доказать, используя непрерывность $f(x)$ и $f_x(x)$, что $\Delta(x)$ — непрерывная функция x . Кроме того, в принятых предположениях $\Delta(x) < 0$ при $x \neq x^*$.

Покажем теперь, что единственной предельной точкой последовательности $\{x^k\}$ (а все $x^k \in R(x^0)$, следовательно, хотя бы одна предельная точка существует) может быть только x^* — точка минимума $f(x)$ в $R(x^0)$. В самом деле, пусть $\tilde{x} (\neq x^*)$ — предельная точка последовательности $\{x^k\}$. Пусть $\Delta(\tilde{x}) = \epsilon < 0$, и пусть в силу непрерывности $\Delta(x) \leq \epsilon/2$ при $\|x - \tilde{x}\| \leq \eta$. В η -окрестности точки \tilde{x} найдется бесконечно много точек из последовательности $\{x^k\}$, причем переход из каждой такой точки в следующую сопровождается уменьшением f не менее, чем на $\epsilon/2$. Поскольку при всех переходах от x^k к x^{k+1} значение f по меньшей мере не возрастает, получено противоречие с предположенной ограниченностью $f(x)$ в области $R(x^0)$. Таким образом, единственной предельной точкой последовательности $\{x^k\}$ является x^* . Теорема доказана.

З а м е ч а н и е. В доказательстве не было использовано то, что шаг спуска s^* и, следовательно $\Delta(x)$, связаны с решением одномерной задачи минимизации (13). Важно лишь то, что $\Delta(x) < 0$ в любой точке, где $f_x(x) \neq 0$, и что $\Delta(x)$ — непрерывная функция. При этом вместо непрерывности может быть использовано и более слабое свойство полунепрерывности: для любого сколь угодно малого $\epsilon > 0$ можно указать такое $\eta > 0$, что из $\|x' - x\| \leq \eta$ следует $\Delta(x') < \Delta(x) + \epsilon$.

Это замечание имеет важные практические последствия, так как им проясняется вопрос о возможной неточности в определении шага спуска s . Например, иногда используется следующий алгоритм вычисления шага. Определим $F(s) \equiv f(x + sy)$ и вычислим $F_s(0) = f_x(x)y$. Обычно $F_s(0) < 0$, что соответствует тому, что y есть направление убывания $f(x)$. Далее назначается некоторый фиксированный «большой» шаг S , и рассматривается прямая L в плоскости (s, F) , проходящая через точку $(0, F(0))$ с наклоном $\frac{1}{S} F_s(0)$ (рис. 70). Если точка $(S, F(S))$ лежит ниже L , то в ка-

честве шага спуска берется S . Если же $F(S) > F(0) - \frac{1}{2} S F_s(0)$, то таким же образом испытывается шаг $S/2$ и т. д., до первого случая, когда

$$F(S/2^{k-1}) > F(0) - F_s(0) S/2^{k-1}, \text{ а } F(S/2^k) < F(0) - F_s(0) S/2^k.$$

Легко видеть, что в изображенной на рис. 70 ситуации все возможные значения шага S , которые могут быть получены при разных $S > s^*$, находятся на отрезке $\left[\frac{1}{2}s^*, s^*\right]$, и такой алгоритм ликвидирует одну из возможных причин несходимости метода спуска: стремление шага спуска к нулю столь быстрое, что бесконечного

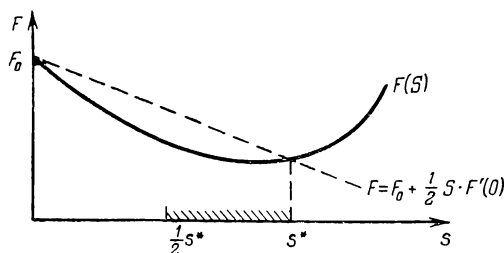


Рис. 70.

числа шагов длиной $s^0, s^1, \dots, s^k, \dots$ не хватает, чтобы от x^0 добраться до x^* . В наших расчетах, представленных в настоящей книге, обычно использовался алгоритм параболической интерполяции, дополненный каким-нибудь простым алгоритмом прерывания итераций. Например, если после получения k -го приближения s_k выполнено неравенство

$$|F(s_k) - F(s_{k-1})| / |F(s_0) - F(s_k)| \leq 0,05 - 0,1, \quad (15)$$

алгоритм прерывался. Число $0,05 - 0,1$ достаточно условно, его удовлетворительность контролировалась еще одним признаком: если задача одномерной минимизации решается точно, то градиенты $f_x(x)$ в двух последовательных точках x^k и $x^{k+1} = x^k + s^k f_x(x^k)$ должны быть ортогональны. Это свойство градиентов проверялось в приближенной форме: считалось нормальным и не требующим уточнения решения задачи определения шага s (13) положение, когда

$$\frac{|(f_x(x^{k-1}), f_x(x^k))|}{\|f_x(x^{k+1})\| \|f_x(x^k)\|} \leq 0,1. \quad (16)$$

Число шагов (т. е. число вычислений функции $f(x^k + sy^k)$), необходимое для такого определения шага спуска s^k , обычно было не очень

большим: от 4 до 7—8 в тех задачах, решение которых было связано с этим алгоритмом.

Алгоритм решения задачи 2 — метод проекции градиента. Перейдем к следующей по сложности задаче

$$\min_{x \in X} f(x), \quad (17)$$

где X предполагается областью столь простой геометрической формы, что операцию проектирования на X можно считать элементарной. Поиск минимума в этом случае осуществляется простым обобщением метода наискорейшего спуска:

1) в точке x^k вычисляется градиент $f_x(x^k) = y^k$;

2) определяется функция скалярного переменного

$$F(s) \equiv f[P_X(x^k - sy^k)]; \quad (18)$$

3) определяется шаг спуска s^k приближенным решением одномерной задачи

$$\min_s F(s); \quad (19)$$

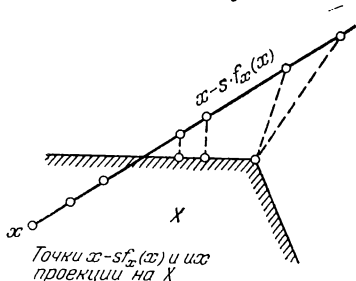


Рис. 71.

4) следующая точка минимизирующей последовательности — $x^{k+1} = P_X(x^k - s^k y^k)$.

Следует, однако, предупредить, что в этом случае, при сколь угодно гладкой зависимости $f(x)$, операция проектирования P_X может привести к тому, что суперпозиция $f(P_X)$ окажется уже негладкой. Такая ситуация показана на рис. 71, где изменение точки $z(s) = P_X(x - sf_x)$, а следовательно, и $f[z(s)]$, содержит «изломы». В этом случае естественнее решать задачу определения s алгоритмом § 46. Параболическая интерполяция может оказаться несходящейся.

Правило множителей Лагранжа, сводящее задачу условной минимизации

$$\min_x f^0(x) \text{ при условиях } f^i(x) = 0, i = 1, 2, \dots, m$$

к задаче на безусловный минимум функции Лагранжа $\mathcal{L}(x, \lambda) \equiv f^0(x) - \sum_{i=1}^m \lambda_i f^i(x)$, хорошо известно. В настоящее время есть много очень изящных доказательств этого правила. Для наших целей будет полезно привести доказательство бесхитрое и прямое, однако имеющее непосредственную связь с алгоритмом построения минимизирующей последовательности точек x^k ,

Пусть дана некоторая точка x , удовлетворяющая условиям $f^i(x)=0, i=1, \dots, m$. Попытаемся проварьировать ее с помощью малого смещения δx так, чтобы, не нарушая (в первом по $\|\delta x\|$ порядке) связей, понизить значение f^0 . Таким образом, получаем задачу: найти δx из условий

$$\delta f^0(\delta x) = f_x^0(x) \delta x < 0, \quad (20)$$

$$\delta f^i(\delta x) = f_x^i(x) \delta x = 0, \quad i = 1, 2, \dots, m. \quad (21)$$

В данной задаче допустимые по условиям (21) δx образуют линейное пространство (в частности, если δx удовлетворяет (21), то и $-\delta x$ удовлетворяет (21), и нам достаточно найти любое δx , удовлетворяющее (21), для которого $f_x^0(x) \delta x \neq 0$). Таким образом, оптимальной (неулучшаемой) точкой x может быть только такая, в которой из $f_x^i(x) \delta x = 0, i=1, 2, \dots, m$ следует $f_x^0(x) \delta x = 0$. В линейной алгебре установлено, что это эквивалентно линейной зависимости градиентов $f_x^i(x)$, т. е. должны существовать множители $\lambda_1, \lambda_2, \dots, \lambda_m$ такие, что

$$f_x^0(x) = \sum_{i=1}^m \lambda_i f_x^i(x) \quad (22)$$

(мы не рассматриваем вырождений типа $f_x^i(x)=0, i=1, 2, \dots, m$). (22) совпадает с условием минимума $\mathcal{L}(x, \lambda)$; $\partial \mathcal{L} / \partial x = 0$. Другая группа условий дает $\partial \mathcal{L} / \partial \lambda_i = f^i(x) = 0$.

Для вычислений важна негативная формулировка теоремы Лагранжа.

Т е о р е м а. Пусть в точке x , удовлетворяющей связям $f^i(x)=0, i=1, \dots, m$, градиенты $f_x^i(x), i=0, 1, \dots, m$, — линейно независимы (т. е. множителей Лагранжа, удовлетворяющих (22), не существует). Тогда в окрестности x может быть найдена точка $x + \delta x$, удовлетворяющая связям и условию $f^0(x + \delta x) < f^0(x)$.

Доказательства в полном объеме проводить не будем, указав лишь на основной момент. Решив задачу (20), (21) и найдя отличный от нуля элемент δx , следует учесть нелинейность задачи, так как $f^i(x + s\delta x) = f_x^i(x) s\delta x + s^2 O(\|\delta x\|^2)$ (из (21), (22) δx находится неоднозначно; будем считать $\|\delta x\|=1$, а s — малый параметр). Далее следует построить поправку y к $s\delta x$, $\|y\|=O(s^2)$, за счет которой связи будут выполнены точно:

$$f^i(x + s\delta x + y) = 0, \quad i = 1, 2, \dots, m$$

и

$$f^0(x + s\delta x + y) = f^0(x) + s f_x^0(x) \delta x + O(s^2) < f^0(x) + \frac{1}{2} f_x^0(x) \delta x.$$

Построение требуемой поправки y проведено в § 5 в более сложной

ситуации. Разумеется, оно опирается на гладкость $f^i(x)$ (например, непрерывность вторых производных).

Алгоритм условной минимизации — метод условного градиента. Следующая по сложности задача — $\min f^0(x)$ при условиях

$$f^i(x) = 0, \quad i = 1, 2, \dots, m. \quad (23)$$

Разумеется, и здесь можно применить метод проекции градиента, но мы считаем (и это соответствует положению дел в прикладных задачах такого сорта), что проектирование на множество X , определяемое системой нелинейных уравнений $f^i(x) = 0, i = 1, 2, \dots, m$, является слишком сложной операцией. Алгоритм поиска условного минимума состоит в том, что для каждой точки x нужно уметь строить «улучшающую» вариацию аргумента δx . При этом приходится иметь в виду не только понижение $f^0(x)$, но и восстановление условий $f^i(x) = 0$, если они оказываются нарушенными. Итак, пусть есть некоторая точка x^k , причем условия $f^i(x^k) = 0$ могут и не выполняться. Считая искомую вариацию δx малой и ограниченной условием $\|\delta x\| \leq S$, где S — шаг процесса, поставим следующую естественную задачу для определения δx :

$$\min_{\delta x} f_x^0(x^k) \delta x \quad (24)$$

при условиях

$$f^i(x^k) + f_x^i(x^k) \delta x = 0, \quad i = 1, 2, \dots, m, \quad (25)$$

$$\|\delta x\| \leq S \quad (\text{или } (\delta x, \delta x) \leq S^2). \quad (26)$$

Эта задача является линеаризацией исходной задачи в S -окрестности точки x^k . Ее решение легко находится методом Лагранжа: образуем функцию

$$\mathcal{L}(\delta x, \lambda) \equiv \lambda_0 f_x^0 \delta x + \sum_{i=1}^m (\lambda_i f_x^i, \delta x) - \frac{1}{2} (\delta x, \delta x), \quad (27)$$

и из условий $\partial \mathcal{L} / \partial x = 0$ найдем выражение для δx :

$$\delta x = \lambda_0 f_x^0 + \sum_{i=1}^m \lambda_i f_x^i. \quad (28)$$

Подставляя это выражение в условия

$$f^i(x^k) + \left(f_x^i, \lambda f_x^0 + \sum_{i=1}^m \lambda_i f_x^i \right) = 0, \quad i = 1, 2, \dots, m, \quad (29)$$

определим λ_i . Для этого нужно дважды решить систему m линей-

ных алгебраических уравнений:

$$1) \quad f^i(x^k) + \sum_{j=1}^m (f_x^i, f_x^j) \lambda_j^* = 0, \quad i = 1, \dots, m; \quad (30)$$

$$2) \quad (f_x^i, f_x^0) + \sum_{j=1}^m (f_x^i, f_x^j) \lambda_j^{**} = 0, \quad i = 1, 2, \dots, m. \quad (31)$$

Общее решение (24) после этого имеет вид $\lambda_j = \lambda_j^* + \lambda_0 \lambda_j^{**}$, а параметр λ_0 находится из условия нормировки

$$\|\delta x\|^2 = \left\| \lambda_0 f_x^0 + \sum_{j=1}^m (\lambda_j^* + \lambda_0 \lambda_j^{**}) f_x^j \right\|^2 = S^2. \quad (32)$$

Это уравнение может быть существенно упрощено с учетом соотношений (25), (26), однако мы этим заниматься не будем, предупредив лишь, что из двух корней (27) нужно отобрать тот, который дает минимум $f_x^0 \delta x$ (второй дает максимум).

З а м е ч а н и е 1. В определенных ситуациях (при больших невязках $f^i(x^k)$) уравнение (27) может и не иметь действительного решения. Это означает, что условия (21) и (22) несовместны. Тогда следует временно отказаться от минимизации f^0 , и целью вариации переменных ($x \rightarrow x + \delta x$) поставить, например, минимизацию $\sum_{i=1}^m [f^i(x^k) + f_x^i \delta x]^2$. После того как в результате нескольких подобных итераций будет получена точка x , удовлетворяющая условиям $f^i(x) = 0$, $i = 1, 2, \dots, m$, начинается собственно процесс решения исходной задачи.

З а м е ч а н и е 2. Во многих редакциях метода условного градиента обычно считается $f^i(x^k) = 0$, $i = 1, \dots, m$ и задача определения δx упрощается. Фактически, поскольку используются конечные вариации δx , в процессе итераций накапливаются невязки в условиях $f^i(x) = 0$, $i = 1, \dots, m$. Когда они достигают некоторой назначенной величины, делается одна или несколько итераций с целью погашения невязок (как это описано выше).

Для того чтобы алгоритм приобрел достаточную четкость, осталось решить еще один важный вопрос — назначение шага S . В данной задаче у нас уже нет естественного критерия для шага: процесс увеличения S сопровождается как падением f^0 , так и ростом невязок в условиях $f^i = 0$. Обозначим через $\delta x(s)$ величину (28), где λ_i определены решением задачи (30), (31), (32), и рассмотрим движение точки $f(x + \delta x(s))$ (в $(m+1)$ -мерном пространстве) при увеличении s от нуля. Предположим для простоты, что $f^i(x) = 0$, $i = 1, \dots, m$. Тогда линия $f(x + \delta x(s))$ ведет себя в окрестности $s = 0$ так, как это качественно показано на рис. 72. Формально невязки $f^i(x + \delta x(s)) = O(s^2)$, $i = 1, 2, \dots, m$. Однако в расчетах

они конечны, и возникает задача оценки: до какой степени увеличение невязок (если оно, разумеется, еще сопровождается уменьшением f^0) может считаться допустимым и не противоречащим основной цели (минимизации f^0 при ограничениях $f^i=0$, $i=1, \dots, m$). Сказанное выше можно уточнить следующим образом: пусть на данной итерации мы выбрали шаг S , приведший к некоторым невязкам. На следующей итерации эти невязки придется компенсировать, что, разумеется, сказывается на величине f^0 . При некоторых невязках, возможно, даже придется ради их погашения пойти на увеличение f^0 . При выборе шага S нужно избежать двух

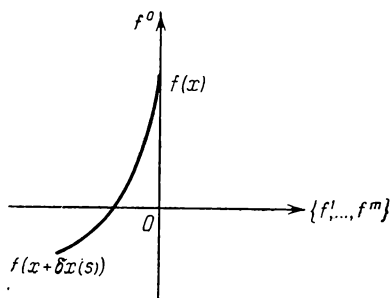


Рис. 72.

крайностей: можно при достаточно малом S сделать процесс очень надежным, так что накопление на каждом шаге невязок, имеющее порядок $O(S^2)$, приведет к нарушению условий $f^1 = \dots = f^m = 0$ на величину $O(S)$ (каждый шаг процесса сопровождается уменьшением f^0 на $O(S)$, следовательно, можно ожидать полного числа шагов $\sim 1/S$). Однако эта слишком осторожная тактика невыгодна, она связана с излишне большим объемом вычислений. Другая

крайность — слишком большие шаги S — также, в конце концов, оказывается неэффективной, так как на одной итерации, получив значительный выигрыш в f^0 за счет нарушения условий, мы будем вынуждены на следующей восстанавливать их, повышая значение f^0 . Итак, нужно иметь какую-то количественную оценку нарушения условий в терминах величины f^0 . Оказывается, такую «цену» определяют, в некотором смысле, множители Лагранжа λ_i . Поясним это, обратившись к рис. 73. С решением задачи (24)–(26) связаны следующие геометрические объекты: сфера вариаций аргумента $\|\delta x\| \leq S$ линейным отображением $\delta f = f_x \delta x$ преобразуется в эллипсоид P в $(m+1)$ -мерном пространстве. Его естественно назвать эллипсоидом смещений. Задача (24)–(26) определяет δx , отображающееся в самую низкую точку эллипсоида смещений, лежащую на оси f^0 . Фактически же точка $f(x + \delta x(s))$ перемещается по кривой Γ , касающейся оси f^0 . (Эту кривую мы более или менее знаем, если в процессе подбора шага вычисляем точки $f(x + \delta x(s_j))$ для нескольких значений j). Выясним, во что обходится нам нарушение условий $f^i=0$. Пусть из каких-то соображений был выбран шаг s^* , и следующий акт варьирования x приводит к задаче (24)–(26), однако уже с новым эллипсоидом, центр которого расположен в точке $f(x + \delta x(s^*))$. Считая $\delta x(s^*)$ малой величиной, мы пренебрежем изменением производных f_x . Тогда новый эллипсоид $P(s^*)$ получается из первоначального P парал-

лельным сдвигом на вектор $f(x + \delta x(s^*)) - f(x) = \Delta f$. Введем следующие характерные точки эллипсоида P : z_0 — нижняя точка эллипсоида P , лежащая по оси f^0 . Именно в эту точку должна была бы переместиться точка $f(x + \delta x(s^*))$, если бы не ошибки линеаризации задачи. Через $z(s)$ обозначим нижнюю точку эллипсоида $P(s)$, лежащую на оси f^0 . В эту точку (в линейном приближении) попадет точка f на следующем шаге процесса минимизации. Нас будет интересовать, как изменяется положение точки $z(s)$ при изменении s и, следовательно, положения центра эллипсоида $P(s)$ на Γ . До тех пор, пока движение центра $P(s)$ по Γ сопровождается движением точки $z(s)$ вниз, мы будем считать, что снижение f^0 на Γ компенсирует увеличение невязок. Точный смысл этого утверждения следующий: два шага процесса, из которых первый переводит точку f из положения $f(x)$ в положение $f(x + \delta x(s)) \in \Gamma$, а второй — в положение $z(s)$ на оси f^0 , приводят к выигрышу в значении f^0 , растущему с ростом s . Однако при переходе через некоторое положение \bar{s} картина меняется, и $z(s)$ при $s > \bar{s}$ начинает перемещаться вверх по оси f^0 ; наступил момент, когда компенсация невязок приводит к снижению эффективности процесса.

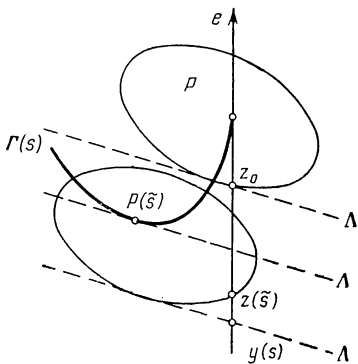


Рис. 73.

Для того чтобы определить \bar{s} , сделаем еще одно приближение: вместо точки $z(s)$ на оси f^0 рассмотрим точку $y(s)$. Она является точкой пересечения оси f^0 с плоскостью Λ , касающейся эллипсоида $P(s)$ в точке $z_0(s)$, получающейся из $z(0)$ сдвигом на Δf . Эта плоскость связана с хорошо известным геометрическим смыслом множителей Лагранжа λ_i , найденных при решении задачи (24)–(26): вектор $\lambda = \{\lambda_0, \lambda_1, \dots, \lambda_m\}$ является нормалью к Λ . Таким образом, увеличение шага s оправдано до тех пор, пока уменьшается величина $(\lambda, f(x + \delta x(s)))$, т. е. функция Лагранжа задачи

$$\mathcal{L}(x, \lambda) = \sum_{i=0}^m \lambda_i f^i(x). \quad (33)$$

Рецепт выбора шага в методе условного градиента. После решения задачи (24)–(26) определяются и множители Лагранжа λ . Образуется одномерная задача определения шага процесса s^* :

$$\min_s \mathcal{L}(x + \delta x(s), \lambda). \quad (34)$$

В качестве следующей точки минимизирующей последовательности берется точка $x + s^* \delta x$.

З а м е ч а н и е 1. Употребление этой методики требует все же известной осторожности: шаг $\|\delta x(s)\|$ не должен быть слишком большим, ведь его обоснование связано с несколькими идеализациями:

1) мы предположили, что эллипсы $P(s)$ не меняются при изменении s , т. е. пренебрегли величинами $s^2 (f_{xx} \delta x, \delta x)$ по сравнению с $s f_x \delta x$; нужно особенно предостеречь вычислителей, пользующихся методом штрафных функций, так как его использование вводит в задачу функции, производные которых сильно меняются при незначительных смещениях точки x ;

2) при оценке результата двух последовательных шагов процесса мы пренебрегли влиянием нелинейности на втором шаге;

3) замена точки $z(s)$ на $y(s)$ оправдана лишь при сравнительно небольших отклонениях Γ от оси f^0 .

З а м е ч а н и е 2. Если ограничения $f^i(x) = 0, i = 1, 2, \dots, m$ сформулированы в терминах линейных функций, ситуация существенно упрощается, кривая Γ лежит на оси f^0 , и для выбора шага s можно без всяких оговорок использовать одномерную задачу $\min f(x + \delta x(s))$.

Пр о б л е м а г л о б а л ь н о г о м и н и м у м а. Все методы, о которых шла речь, если сходятся, то лишь к точке локального минимума функции. Если таких точек несколько, результат зависит от выбора начального приближения. Разумеется, хотелось бы получить абсолютный минимум, да еще и гарантию, что получен действительно абсолютный (глобальный) минимум. К сожалению, в общем случае эта задача, видимо, неразрешима. Точнее, практически неразрешима. Единственный реальный подход к этой задаче состоит в том, чтобы, начиная из разных начальных точек, алгоритмом спуска найти возможно большее число точек локального минимума и отобрать из них точку с наименьшим значением функции. Что касается выбора стартовых точек спуска, то при отсутствии каких-то частных, связанных с данной задачей содержательных указаний, приходится выбирать их случайным образом. К сожалению, в сложных прикладных задачах поиск локального минимума сам по себе достаточно трудоемок, так что возможности комбинировать его со случайным выбором стартовых точек весьма ограничены. Это, разумеется, относится и к задачам оптимального управления, причем ситуация осложняется еще и тем, что аргументом является функция, и выбор даже не очень плотного множества стартовых точек в функциональном пространстве с последующим спуском приводит к нереальным затратам машинного времени. Поэтому приходится ограничиваться сравнительно небольшим числом проб, по возможности используя для

выбора начальных приближений максимум содержательной информации о задаче. В литературе, тем не менее, время от времени появляются работы, «решающие» проблему глобального минимума. По мнению автора, большая часть таких работ основана на непонимании следующего простого факта: эта проблема совершенно тривиальна, если мы не принимаем во внимание объем вычислений, поэтому можно в изобилии генерировать формальные алгоритмы ее решения. В действительности, они не сдвигают дело ни на шаг. Вот примеры таких «решений».

1. Поиск $\max_{x \in X} f(x)$ (X — замкнутое ограниченное множество) можно свести к вычислению $\left\{ \int_X f^p(x) dx \right\}^{1/p}$; для определения точки минимума нужно еще вычислить $\left\{ \int_X x f^p(x) dx \right\}$. А для вычисления

многомерных интегралов, как известно, разработаны эффективные методы (следуют ссылки на действительно обширную и солидную литературу по методу Монте-Карло).

Интегрирование методом Монте-Карло осуществляется вычислением значений функции $f(x)$ в большом числе «случайных» точек, и в качестве интеграла берется некоторое среднее значение всех $f(x_i)$. Если идти по этому пути, то проще и надежнее взять в качестве приближенного значения $\max f(x)$ наибольшее из вычисленных значений и соответствующую ему точку x_i (впрочем, если угодно, можно получить этот же рецепт с помощью интеграла, взяв $p = \infty$).

2. Можно построить динамическую гамильтонову систему, взяв $f(x)$ в качестве потенциала. Траектория этой системы, начинающаяся в точке $\{x^0, \dot{x}^0\}$, в силу эргодичности рано или поздно пересечет любую ϵ -окрестность каждой точки фазового пространства, в которой потенциальная энергия $f(x)$ не больше начальной энергии $f(x^0) + \frac{1}{2}(\dot{x}^0)^2$. Теперь остается численно интегрируя систему дифференциальных уравнений, одновременно определять на ней $\min_t f[x(t)]$. Если иметь в виду с помощью этого метода просмотреть, так сказать, некоторую часть фазового пространства, то это можно сделать и проще, с помощью случайного равномерно распределенного в X набора точек x . Пожалуй, единственный довод в пользу отслеживания траектории динамической системы состоит в том, что при этом будет рассмотрено не все множество X , а только та его часть, в которой $f(x) \leq f(x^0) + \frac{1}{2}(\dot{x}^0)^2$. При этом \dot{x}^0 нужно брать таким, чтобы система могла преодолеть разделяющие точки локальных минимумов потенциальные барьеры. Однако нет серьезных оснований (ни теоретических, ни экспериментальных) считать этот подход более надежным и эффективным, чем комбинация

метода спуска со случайным выбором в X стартовых точек. Под эффективностью следует здесь понимать следующее: пусть на определение $\min f(x)$ выделено N вычислений $f(x)$. Более эффективным следует считать метод, который за N «испытаний» даст точку с меньшим значением $f(x)$.

О скорости сходимости метода спуска по градиенту. В общем случае трудно оценить скорость сходимости процесса наискорейшего спуска, каждый шаг которого состоит в решении задачи $\min_s f(x - sf_x)$. Однако определенную

информацию на этот счет можно получить, предположив $f(x)$ положительно-определенной квадратичной формой $f(x) = (Ax, x)$. Такой анализ достаточно полно отражает свойства наискорейшего спуска в окрестности минимума, где f уже можно аппроксимировать двумя членами ряда Тейлора. Анализ показывает, что расстояние до точки минимума убывает как $\left(1 - 2 \frac{\lambda_{\min}}{\lambda_{\max}}\right)^k$, где k — число шагов спуска,

λ_{\min} , λ_{\max} — минимальное и максимальное собственные числа матрицы вторых производных A (A , очевидно, симметрична). Величины λ_{\min} и λ_{\max} имеют простой геометрический смысл. Рассмотрим в малой окрестности точки минимума $f(x)$ поверхности равного уровня $f(x) = c$. Если пренебречь членами выше второго порядка, эти поверхности суть эллипсоиды. При этом отношение $\lambda_{\min}/\lambda_{\max}$ характеризует эксцентриситет этого эллипсоида. Наиболее неблагоприятной и трудной является ситуация, когда эти эллипсоиды сильно вытянуты. Такие ситуации характеризуют, например, термином «овраг». Можно еще иначе характеризовать «овражные» функции: это функции, дифференциальные свойства которых очень сильно различаются в разных направлениях. Уточним, что это значит. Рассмотрим функции $F(e, s) \equiv f(x + se)$, где x — некоторая точка, s — скаляр, а e — векторный параметр, определяющий направление ($\|e\| = 1$). Таким образом, мы имеем семейство функций $F(e, s)$ одного переменного s . Константы, ограничивающие производные по s этих функций, зависят от e , и если при разных e они имеют существенно разные величины, мы характеризуем функцию как «овражную», трудную для процессов поиска минимума.

О методе тяжелого шарика. Стремясь повысить эффективность метода спуска по градиенту, было предложено (см. [35], [60]) использовать траекторию тяжелого шарика, движущегося с трением. Эта траектория определяется уравнением $\ddot{x} + f_x(x) + \lambda \dot{x} = 0$ (λ — коэффициент трения). В [35] приводятся качественные, основанные на механической аналогии, соображения в пользу преимущества этой системы перед системой уравнений градиентного спуска $\dot{x} = -f_x(x)$. Однако, на таком уровне аргументации столь же убедительно можно утверждать и обратное. Объективный анализ возможностей метода тяжелого шарика можно

провести, опять-таки предположив $f(x)$ квадратичной формой (Ax, x) . Тогда численное интегрирование по разностной схеме

$$\frac{x^{k+1} - 2x^k + x^{k-1}}{\tau^2} + \lambda \frac{x^{k+1} - x^{k-1}}{2\tau} + 2Ax = 0$$

оказывается эквивалентным процессу минимизации формы (Ax, x) так называемым двухшаговым процессом. Анализ сходимости может быть проведен (см., например, [60]), и результат следующий: при специальном (и достаточно аккуратном) выборе λ и τ расстояние до точки минимума убывает, как $(1 - 2\sqrt{\lambda_{\min}/\lambda_{\max}})^k$. Видно, что сходимость существенно более быстрая, чем в методе наискорейшего спуска, но выбор τ и λ требует знания границ спектра: λ_{\min} , λ_{\max} . Если же τ и λ выбирать как-нибудь, то можно не получить никакого выигрыша по сравнению с методом наискорейшего спуска, и даже получить худший результат. А ведь «механическая» аргументация не зависит от значений λ и τ , что и показывает ее истинную цену.

§ 46. Поиск минимума. Негладкие задачи

Вычислительные методы предназначены прежде всего для решения задач, возникающих в приложениях. Авторами таких задач являются инженеры, физики, медики и т. д., т. е. специалисты, не искушенные в изобретении хитроумных примеров функций, не имеющих, например, производной нигде, и т. д. Для таких специалистов термины «функция» и «формула» (имеется в виду формула не очень сложная) — практически равнозначны. Поэтому, на первый взгляд, от них не следует ожидать задач с недифференцируемыми функциями. Однако это не так. Есть две весьма популярные в приложениях операции, с помощью которых из сколь угодно гладких функций образуются негладкие. Это операции \max и $|\cdot|$. Вычислитель должен быть готов к задачам минимизации функций

$$F(x) \equiv \max f^i(x), \quad (1)$$

$$F(x) \equiv \sum_i |f^i(x)|. \quad (2)$$

При этом обычно $f^i(x)$ — достаточно гладкие. Однако $F(x)$ уже не являются всюду дифференцируемыми, хотя они и не совсем уж скверные. Функции $F(x)$ (1), (2) принадлежат к важному в приложениях классу функций, которые в каждой точке x дифференцируемы по направлениям. Это означает, что для любого вектора y (той же размерности, что и x) существует

$$\lim_{\substack{\varepsilon \rightarrow 0 \\ \varepsilon > 0}} \frac{F(x + \varepsilon y) - F(x)}{\varepsilon} = D(x, y).$$

Величина $D(x, y)$ называется *производной* $F(x)$ в точке x по направлению y . Почти во всякой точке x функции $F(x)$ (1), (2) имеют $F(x)$ обычную производную (т. е. $D(x, y) = (g, y)$, где $g = \text{grad}$). Функция $F(x)$ (1) не имеет обычной производной только в том случае, если по крайней мере для двух индексов i_1, i_2 $f^{i_1}(x) = f^{i_2}(x) = F(x)$. Функция $F(x)$ (2) не имеет обычной производной лишь в тех точках x , в которых $f^{i_1}(x) = 0$ хотя бы для одного индекса i_1 . Однако при решении задачи $\min_x F(x)$,

как правило, приходится иметь дело именно с этими исключительно редкими точками x . Поиск минимума негладких функций $F(x)$ осуществляется по той же общей схеме построения минимизирующей последовательности точек. Пусть имеется некоторая точка x , для нее тем или иным способом строится направление спуска y , и следующая точка берется в виде $x + sy$, где s — шаг спуска, определяемый, например, решением одномерной задачи

$$\min_s F(x + sy) \equiv \min \varphi(s). \quad (3)$$

Однако теперь обе задачи — найти y и решить задачу (3) — существенно усложняются. Начнем с задачи (3). Для ее решения разработаны алгоритмы последовательных испытаний, т. е. вычислений $\varphi(s)$ в некоторых специально выбираемых точках s_k , имеющие целью локализацию точки минимума ценой наименьшего числа испытаний. Но эти алгоритмы работают только при определенных предположениях. Пусть ищется минимум функции $\varphi(s)$ на интервале $[0, 1]$, причем на этом интервале $\varphi(s)$ — унимодальная функция, т. е. имеющая единственную точку минимума s^* . На интервале $[0, s^*]$ φ монотонно убывает, на $[s^*, 1]$ — монотонно возрастает; непрерывность φ не предполагается, s^* может совпадать с любым концом интервала $[0, 1]$. Пусть на приближенное определение точки минимума $\varphi(s)$ выделено определенное число n испытаний. Ставится задача так выбрать точки $\{s_1, s_2, \dots, s_n\}$ последовательных испытаний, чтобы после их проведения можно было указать *интервал локализации* $[\alpha, \beta] \subset [0, 1]$, на котором заведомо находится точка минимума φ . Обозначим длину интервала локализации после n испытаний через $l_n(s_1, s_2, \dots, s_n, \varphi)$. Тогда задача наилучшей организации испытаний φ , т. е. использования результатов уже проделанных вычислений при выборе очередной точки s_k , может быть формализована следующим образом:

$$\min_{P_1, P_2, \dots, P_n} \max_{\varphi \in \Phi} l_n[P_1, P_2(s_1, \varphi), \dots, P_n(s_1, s_2, \dots, s_{n-1}, \varphi), \varphi], \quad (4)$$

где Φ — множество всех унимодальных на $[0, 1]$ функций, а $P_1, P_2(s_1, \varphi_1), P_3(s_1, s_2, \varphi_1, \varphi_2)$ — некоторые правила (функции), вычисляющие точки испытаний s_1, s_2, \dots, s_n в зависимости от уже

полученных результатов. Хотя задача (4) выглядит устрашающе, она решена так называемым алгоритмом Кифера. Мы не будем его излагать, так как практически более удобен другой алгоритм (метод золотого сечения). Однако сначала обсудим совсем тривиальную идею — метод последовательного деления интервала локализации пополам.

Лемма 1. Пусть так или иначе получен некоторый интервал $[\alpha, \beta]$ локализации точки минимума ($s^* \in [\alpha, \beta]$). Тогда с помощью двух испытаний его длина может быть сокращена вдвое.

В самом деле, вычислим φ в точках $s' = \frac{\alpha + \beta}{2} - \varepsilon$, $s'' = \frac{\alpha + \beta}{2} + \varepsilon$, где $\varepsilon > 0$ — очень малая величина. Тогда, если $\varphi(s') < \varphi(s'')$, то $s^* \in [\alpha, s'']$; при $\varphi(s'') < \varphi(s')$ $s^* \in [s', \beta]$ (случай $\varphi(s') = \varphi(s'')$, при котором s^* локализуется на отрезке $[s', s'']$ — не рассматриваем). Так что, строго говоря, интервал локализации сократился не вдвое, он стал $\frac{\beta - \alpha}{2} + \varepsilon$, но величиной ε мы пренебрежем. Таким образом, в результате n испытаний первоначальный интервал локализации $[0, 1]$ сократится до размеров $\sim \left(\frac{1}{2}\right)^{n/2}$.

Алгоритм золотого сечения при том же числе испытаний дает более точную локализацию точки минимума. Обозначим через $[\alpha_0, \beta_0]$ первоначальный интервал локализации (в нашем случае $[0, 1]$) и проведем испытания в двух точках $a_1 < b_1$, причем a_1 и b_1 расположены симметрично относительно середины интервала $\frac{\alpha_0 + \beta_0}{2}$, и каждая из точек a_1 и b_1 производит так называемое «золотое сечение» отрезка $[\alpha_0, \beta_0]$, т. е. имеют место соотношения

$$\frac{\beta_0 - \alpha_0}{\beta_0 - a_1} = \frac{\beta_0 - a_1}{a_1 - \alpha_0}, \quad \frac{\beta_0 - \alpha_0}{\beta_1 - \alpha_0} = \frac{b_1 - \alpha_0}{\beta_0 - b_1}; \quad b_1 - \alpha_0 = \beta_0 - a_1. \quad (5)$$

Сравнением значений $\varphi(a_1)$ и $\varphi(b_1)$ мы можем сократить интервал локализации, выбирая в качестве $[\alpha_1, \beta_1]$ либо $[\alpha_0, b_1]$ (при $\varphi(a_1) < \varphi(b_1)$), либо $[a_1, \beta_0]$ (при $\varphi(b_1) < \varphi(a_1)$); случай $\varphi(a_1) = \varphi(b_1)$ не рассматривается как маловероятный и слишком удачный: при этом $[\alpha_1, \beta_1] = [a_1, b_1]$. В обоих случаях длина интервала локализации сократилась одинаково: $(\beta_1 - \alpha_1) = \frac{\sqrt{5} - 1}{2} (\beta_0 - \alpha_0) \approx 0,618 (\beta_0 - \alpha_0)$. И в обоих случаях на новом отрезке $[\alpha_1, \beta_1]$ уже есть точка (a_1 или b_1), в которой проведено испытание и которая также осуществляет золотое сечение отрезка $[\alpha_1, \beta_1]$. Легко проверить, что из (5) следует, например (для $[\alpha_1, \beta_1] = [\alpha_0, b_1]$),

$$\frac{b_1 - \alpha_0}{a_1 - \alpha_0} = \frac{a_1 - \alpha_0}{b_1 - a_1}, \quad \text{т. е.} \quad \frac{\beta_1 - \alpha_1}{a_1 - \alpha_1} = \frac{a_1 - \alpha_1}{\beta_1 - a_1}.$$

В качестве следующей точки испытания берется вторая точка, осуществляющая золотое сечение $[\alpha_1, \beta_1]$ — она симметрична

первой относительно точки $\frac{\alpha_1 + \beta_1}{2}$. Таким образом, ценой еще одного вычисления φ мы сможем сократить интервал неопределенности до размеров

$$(\beta_2 - \alpha_2) \approx 0,618 (\beta_1 - \alpha_1) \approx (0,618)^2 (\beta_0 - \alpha_0).$$

Далее процесс продолжается до получения нужной точности. Подведем итог.

Т е о р е м а 1. *Алгоритм золотого сечения позволяет ценой n вычислений функции φ довести интервал локализации точки минимума $\varphi(s)$ до размера $(0,618)^{n-1} (\beta_0 - \alpha_0)$.*

Опущенный при анализе случай $\varphi(a_1) = \varphi(b_1)$ приводит к интервалу локализации $[\alpha_1, \beta_1]$ длиной $0,236 (\beta_0 - \alpha_0)$. В этой ситуации следует начинать процесс как бы с начала; если такие ситуации будут встречаться, они только улучшат оценку. Таким образом, алгоритм золотого сечения лучше алгоритма деления интервала пополам; в последнем длина интервала локализации

$$\left(\frac{\sqrt{2}}{2}\right)^n (\beta_0 - \alpha_0) \approx 0,71^n (\beta_0 - \alpha_0).$$

Например, при 20 испытаниях алгоритм золотого сечения даст интервал локализации примерно в 12—13 раз меньший, чем алгоритм деления пополам. Выше упоминался оптимальный алгоритм Кифера. Используя его, вычислитель не получит существенного выигрыша: интервал локализации уменьшится (по сравнению с тем, что дал алгоритм золотого сечения) разве лишь на 2—3%. Таким образом, алгоритм Кифера имеет в основном теоретическое значение, показывая, что алгоритм золотого сечения практически оптимален.

Н а п р а в л е н и е с п у с к а . М е т о д Н ь ю т о н а . Перейдем к более сложной задаче — определению направления y , вдоль которого функция $F(x)$ убывает. Рассмотрим сначала функцию $F(x) = \max f^i(x)$. Обозначим через M_ϵ множество индексов i , для которых $f^i(x)$ «почти максимально»:

$$\{i \in M_\epsilon \text{ экв. } f^i(x) \geq F(x) - \epsilon, \quad \epsilon > 0\}. \quad (6)$$

Через I_ϵ обозначим число входящих в M_ϵ индексов (назначение ϵ обсуждается ниже). Определим величину смещения δx точки x решением следующей задачи, являющейся линеаризацией исходной:

$$\min_{|\delta x| \leq S} \max_{i \in M_\epsilon} [f^i(x) + f_x^i(x) \delta x]. \quad (7)$$

Здесь S — некоторое число, шаг процесса. Задача (7) является сравнительно стандартной задачей линейного программирования

и может быть решена соответствующим алгоритмом. Особая точность в ее решении не нужна, поскольку сама задача — приближенная. Определив δx , можно поступать двояко: либо считать δx искомым смещением и перейти к точке $x + \delta x$, либо считать δx направлением спуска и перейти к точке $x + s\delta x$, определив скаляр s решением одномерной задачи $\min_s F(x + s\delta x)$. Второй путь надежнее, но требует больших вычислительных затрат на один шаг. В своей вычислительной практике автор обычно использовал первый способ, дополняя его некоторым простым алгоритмом регулирования величины S . Основным критерием, указывающим на необходимость уменьшения или увеличения, было сравнение предсказанного значения $F(x + \delta x)$

$$F_{\text{пр}}(x + \delta x) = \max_i [f^i(x) + f_x^i(x)\delta x]$$

с фактическим значением $F(x + \delta x)$. Что касается величины ϵ , то существуют две крайности, которых следует избегать.

1. Если ϵ слишком мало, то в M_ϵ не войдет индекс k , для которого $f^k(x)$ почти равно $F(x)$. При решении задачи (7) «интересы» функции f^k не будут учтены, найденное направление δx может оказаться направлением роста $f^k(x)$, поэтому при движении x по лучу $x + s\delta x$ F будет уменьшаться лишь при очень малых s , до тех пор, пока $\max_i f^i(x + s\delta x)$ не станет определяться растущей функцией $f^k(x + s\delta x)$.

2. Если ϵ слишком велико, в M_ϵ войдет много «лишних» индексов i , что может затруднить решение задачи (7). Однако в целом следует больше опасаться слишком малых ϵ , чем больших. Можно предложить и практический критерий, позволяющий судить о том, является ли назначенная величина ϵ слишком малой и не следует ли ее увеличить. Пусть на каком-то шаге получена точка x , $F(x) = f^k(x)$, и пусть на предыдущем шаге индекс k не входит в M_ϵ . Это означает, что ϵ слишком мало и его следует увеличить. Ниже мы подробнее рассмотрим пример решения подобной задачи.

При решении задач типа

$$\min_x \sum_i |f^i(x)|$$

возникают аналогичные проблемы. Здесь метод Ньютона приводит к следующим вычислениям. Выделяются подмножества индексов M_ϵ^+ , M_ϵ^- , M_ϵ^0 :

$$\begin{aligned} i \in M_\epsilon^+ & \quad \text{экв.} \quad f^i(x) > \epsilon, \\ i \in M_\epsilon^- & \quad \text{экв.} \quad f^i(x) < -\epsilon, \\ i \in M_\epsilon^0 & \quad \text{экв.} \quad |f^i(x)| \leq \epsilon, \end{aligned}$$

и смещение δx точки x находится решением задачи

$$\min_{|\delta x| \leq \epsilon} \left\{ \sum_{M_i^+} f_x^i \delta x - \sum_{M_\epsilon^-} f_x^i \delta x + \sum_{M_\epsilon^0} |f^i(x) + f_x^i \delta x| \right\}. \quad (8)$$

Это тоже, в сущности, задача линейного программирования (см. § 47). Выбор величины ϵ определяется двумя соображениями, как всегда предъявляющими к ϵ противоположные требования: ϵ должно быть по возможности меньше, так как трудность задачи (8) связана с числом индексов в M_ϵ^0 . С другой стороны, ϵ не должно быть слишком малым: не следует допускать перехода за один шаг процесса какого-то индекса i из M_ϵ^- , например, в M_ϵ^+ .

Метод обобщенного градиента для решения негладких задач оптимизации. Одной из мощных тенденций в разработке методов решения задач оптимизации является стремление к возможно более простой (внешне) формулировке вычислительной задачи. В частности, все многообразие задач может быть приведено к простейшей форме

$$\min_x F(x). \quad (9)$$

Разумеется, эта упрощенная задача не эквивалентна исходной, но лишь аппроксимирует ее с любой необходимой точностью. Здесь открывается соблазнительная возможность унифицированного подхода как при разработке алгоритмов, так и при создании набора стандартных программ. К сожалению, эта внешняя простота не дается даром. Сведение сложной задачи к «простой» (9) достигается за счет резкого ухудшения дифференциальных свойств $F(x)$ по сравнению с дифференциальными свойствами функций исходной содержательной постановки задачи. Заметим, что под дифференциальными свойствами вычислитель должен понимать не столько словесные характеристики типа «непрерывная функция», «дифференцируемая», «дважды дифференцируемая» и т. д., сколько величины констант в характеристиках непрерывности, дифференцируемости. Поэтому тот факт, что методом «штрафных функций» можно свести общую задачу оптимизации к задаче (9) даже с бесконечно дифференцируемой $F(x)$, не следует переоценивать. Рассмотрим характерную для упомянутой тенденции попытку решать задачу (9) с функцией

$$F(x) = \max_i f^i(x) \quad (10)$$

простыми и привычными вычислительными средствами, не требующими обращения к таким неприятным вещам, как алгоритмы линейного программирования. Причем, что тоже характерно, этот подход подкреплен соответствующими теоремами о сходи-

мости. Функция (10), как уже отмечалось, не имеет градиента в обычном смысле слова, однако в [82] предложено использовать так называемый *обобщенный градиент* (О. Г.). Не вдаваясь в излишнюю аккуратность, поясним, что такое О. Г. Пусть M — множество индексов i , для которых $f^i(x) = F(x)$. Тогда обобщенным градиентом $F(x)$ является обычный градиент любой из таких функций f^i : $g = f_x^i(x)$, $i \in M$. Предлагается такой обобщенный градиент g использовать в качестве направления спуска $F(x)$. На первый взгляд, это — странное предложение: ведь если число входящих в M индексов $I > 1$, функция $\varphi(s) \equiv F(x - sg)$ может расти как при $s > 0$, так и при $s < 0$. Тем не менее движение по О. Г. при определенных предположениях о функциях $f^i(x)$ (например, если они выпуклы) является движением в сторону искомой точки минимума. Поясним это рис. 74, на котором для $I=2$ изображены линии уровня двух функций $f^1(x)$, $f^2(x)$, данная, варьируемая точка x , искомая точка минимума x^* и градиенты g_1 , g_2 , каждый из которых может быть взят в качестве О. Г. g . Оба направления, g_1 и g_2 , составляют острый угол с направлением xx^* , и продвижение по каждому из них при малых s сопровождается убыванием $\|x - sg - x^*\|$. Однако, и это тоже характерная

деталь, у нас уже нет разумного критерия для выбора шага s : ведь и то, и другое направления g являются направлениями роста $F(x - sg)$. Теоретика это не смутит, у него давно готов ответ: нужно взять в качестве последовательных шагов спусков числа

$s_k > 0$ такие, что $s_k \rightarrow 0$ и $\sum_{k=1}^{\infty} s_k = \infty$. Однако этот классический

рецепт практически мало полезен; он, в сущности, лишь дает уверенность в том, что в принципе задача разрешима. Попытки применения этого метода сразу же показали его крайне медленную сходимость и ненадежность результатов: можно пояснить, в каких ситуациях метод движения по О. Г. становится ненадежным. Подобно тому как обычный градиент дифференцируемой функции определяет гиперплоскость, отделяющую направления роста функции от направлений ее убывания (и именно последние представляют наибольший интерес с точки зрения построения минимизирующих последовательностей точек), так и для минимизации функции $F(x)$ важен выпуклый конус K , определяемый

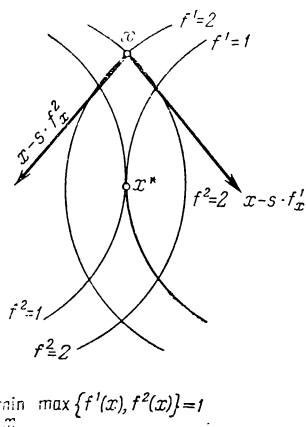
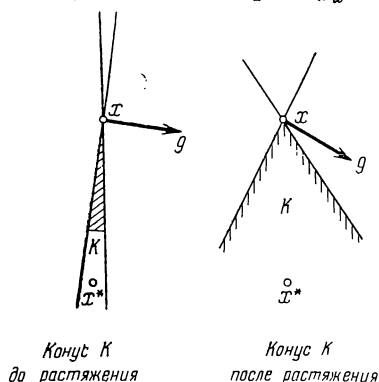


Рис. 74.

СООТНОШЕНИЯМИ

$$\{y \in K(x) \text{ экв. } (f'_x(x), y) \leq 0, i \in M\}. \quad (11)$$

Внутренность этого конуса K есть совокупность направлений убывания F . Для метода движения по О. Г. наиболее трудны ситуации, в которых конус K становится очень узким. Это происходит при малых значениях I в ситуации, близкой к оптимальной, а при больших значениях I и вдалеке от минимума. Кстати, необходимым признаком оптимальности является вырождение конуса K — его внутренность пуста. Нетрудно понять, что при этом совокупность векторов $\{f'_x, i \in M\}$ оказывается линейно зависи-



[Рис. 75.]

мой. Как правило, это наступает в ситуации, когда число входящих в M индексов I сравнивается с размерностью пространства x , хотя, в принципе, не исключена линейная зависимость векторов $\{f'_x, i \in M\}$ и при I , меньшем размерности x . Объяснение медленной сходимости метода О. Г. узостью конуса K послужило основой для одного из методов ускорения его сходимости. Этот метод, предложенный и разработанный Шором [83], основан на подходящем преобразовании пространства x с тем, чтобы в новых переменных конус K стал шире. Это достигается операциями последовательного растяжения пространства x по направлениям последовательных О. Г. Суть дела поясняет рис. 75, на котором изображен (для $I=2$) конус K до и после растяжения пространства в направлении g . Операция растяжения не вполне детерминирована — остается произвол в выборе коэффициента растяжения. Так или иначе, этот прием был отработан, усложнен растяжением в направлении разности двух последовательных градиентов, что в совокупности с некоторой техникой подбора шагов движения по О. Г. существенно повысило эффективность и надежность метода О. Г. Читатель, может познакомиться с подробностями по работам [83], [84]. Здесь мы этих деталей не излагаем, поскольку автор не является сторонником подобных методов, полагая, что вычислительные методы, явно использующие достаточно полный анализ конуса K , должны быть более эффективными. Метод Ньютона как раз и основан на анализе конуса K . Для подтверждения этой точки зрения мы сейчас проведем сравнение решения некоторой модельной задачи методом обобщенного градиента с растяжением пространства и методом Ньютона.

Модельная задача, которая будет решаться, приведена в [84]. Она имеет вид (10), причем

$$f^i(x) = A_i \sum_{j=1}^n (x_j - a_{ij})^2, \quad i = 1, 2, \dots, m (n=5, m=10)$$

$$A_i = \{1; 5; 10; 2; 4; 3; 1,7; 2,5; 6; 3,5\}.$$

$$a_{ij} = \begin{pmatrix} 0 & 2 & 1 & 1 & 3 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 2 & 4 & 2 & 2 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 \\ 0 & 1 & 1 & 2 & 0 & 0 & 1 & 2 & 1 & 0 \\ 0 & 3 & 2 & 2 & 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Сначала приведем результаты решения этой задачи методом спуска по обобщенному градиенту. Следующая ниже таблица 1

Т а б л и ц а 1 (заимствована из [84])

k	0	1	6	11	16	21	26
F	80	63,09	34,40	25,84	24,639	22,782	22,650
k	31	36	41	46	51	F_{\min}	
F	22,609	22,604	22,6017	22,60064	22,60023	22,60016	

иллюстрирует убывание функции $F(x) = \max_i f^i(x)$ с номером шага спуска k . Спуск начинался из точки

$$x = \{0, 0, 0, 0, 1\}.$$

Подчеркнем, что этот результат был получен при достаточно хитроумной технике последовательного растяжения пространства. Его следует признать очень удачным. Подробнее мы опишем решение этой же задачи методом Ньютона (7). Этот метод был дополнен простым алгоритмом изменения шага S : если после перехода от x к $x + \delta x$ оказывалось $F(x + \delta x) > F(x)$, происходил возврат к x , и снова определялось смещение δx решением задачи линейного программирования (7), но уже с меньшим (примерно в 2,5 раза) S . В случае $F(x + \delta x) < F(x)$, шаг S не менялся. Процесс решения задачи показан в табл. 2. В ней представлены следующие величины:

k — номер шага, предсказанное для этого k значение $F(x + \delta x)$ (на основе линеаризации), фактическое значение $F(x + \delta x)$,

Таблица 2

k	F предск.	F факт.	f^2	f^3	f^4	f^5	f^6	S	n
0	—	80,0000	55,00	80,00	46,00	56,00	36,00		
1	32,00	36,187	28,09	36,187	32,384	36,082	33,712	0,50	1
2	20,493	26,683	21,941	17,05	22,557	23,091	26,683	0,50	1
3	20,259	24,057	23,424	22,578	21,525	22,791	24,057	0,50	1
4	22,335	22,970	22,852	20,659	22,508	22,784	22,970	0,21	2
5	22,297	22,889	22,794	19,131	22,381	22,752	22,889	0,21	1
6	22,630	22,715	22,599	19,730	22,536	22,715	22,675	0,088	2
7	22,538	22,648	22,648	21,188	22,565	22,593	22,648	0,088	1
8	22,586	22,6076	22,6040	20,634	22,593	22,6003	22,6076	0,037	3
9	22,6015	22,6039	22,5998	20,547	22,5961	22,6039	22,6009	0,016	2
10	22,5994	22,6032	22,6025	20,648	22,6007	22,6019	22,6032	0,016	1
11	22,597	22,6011	22,6005	20,554	22,5987	22,5999	22,6011	0,016	1
12	22,5999	22,60056	22,60046	20,594	22,60013	22,60034	22,60056	0,0065	2
13	22,5997	22,60039	22,60028	20,56	22,59995	22,60017	22,60039	0,0065	1
14	22,60010	22,60021	22,60019	20,57	22,60013	22,60017	22,60021	0,0011	3

значения функций $f^i(x)$, $i \in M_*$, шаг S и число n решений задачи (7), потребовавшееся для перехода от x к $x + \delta x$ (при $n=1$ первый же переход от x к $x + \delta x$ сопровождается падением F , при $n=2$ первый переход привел к росту F , второй с уменьшенным S , — к падению F , и т. д.). В этом расчете подбор шага S был совсем бесхитростным. Попробуем его несколько усложнить, используя в качестве критерия сравнение предсказанного на основе линейной теории $F_n(x + \delta x)$ с фактическим $F(x + \delta x)$.

Используем следующие формулы:

$$F_n(x + \delta x) \simeq F(x) + AS,$$

$$F(x + \delta x) \simeq F(x) + AS + BS^2.$$

Имея $F_n(x + \delta x)$ и $F(x + \delta x)$, можно вычислить по очевидным формулам коэффициенты A и B , после чего находится «наилучшее» значение $S_{\text{опт}} = -A/2B$. Следующий шаг процесса осуществляется с шагом $S = 0,8 S_{\text{опт}}$ ($0,8$ — коэффициент «осторожности»). Как и в первом случае, при $F(x + \delta x) > F(x)$ происходит возврат к старому x . Результат решения показан в табл. 3 теми же величинами, что и в табл. 2. В табл. 3 представлены и результаты неудачных шагов, сопровождающихся ростом F ; их номера отмечены *. Видно, что используемая процедура подбора шага — груба (это можно было предвидеть и заранее), так что почти каждый второй шаг процесса минимизации — неудачен и служит лишь для определения приемлемого шага S . Заметим еще, что задача оказалась вырожденной: в ее решении $\max f^i(x)$ достигается не при пяти индексах, как должно быть в общем случае,

Т а б л и ц а 3

k	F предск.	F факт.	f^2	f^3	f^4	f^5	f^6	S
0	—	80,00	55,00	80,00	46,00	56,00	36,00	
1	32,00	36,19	28,09	36,19	32,38	36,08	33,71	0,5
2*	3,3	50,61	45,04	50,61	19,99	19,44	46,08	1,5
2	19,91	27,05	21,70	17,00	22,29	21,78	27,05	0,52
3*	20,55	27,26	24,63	27,26	22,32	23,08	24,71	0,60
3	21,98	23,08	22,86	20,46	22,26	22,75	23,08	0,29
4*	20,97	27,73	26,60	23,19	23,23	25,46	27,73	0,66
4	22,45	22,678	22,618	19,94	22,507	22,611	22,678	0,13
5*	22,46	23,30	22,907	23,30	22,641	22,819	22,996	0,18
5	22,583	22,606	22,602	20,630	22,591	22,598	22,606	0,036
6*	22,594	22,683	22,668	19,644	22,624	22,654	22,683	0,075
6	22,599	22,6026	22,6021	20,42	22,6003	22,6015	22,6027	0,015
7	22,598	22,60126	22,6007	20,64	22,5989	22,6001	22,6013	0,015
8*	22,600	22,60127	22,6010	20,49	22,60012	22,6007	22,60127	0,010
8	22,6000	22,60040	22,6003	20,57	22,60012	22,60026	22,60040	0,005

а лишь при четырех: $M = \{2, 4, 5, 9\}$. Однако при $\epsilon = 2 \div 10$, $M_\epsilon = \{2, 3, 4, 5, 9\}$; именно это множество и было использовано в расчетах. Алгоритм подбора шага себя оправдал, но к существенному выигрышу в эффективности не привел. Что касается сравнения эффективности метода Ньютона и метода спуска по обобщенному градиенту, то в данной задаче метод Ньютона все же эффективнее, хотя о значительном преимуществе говорить нельзя. Видимо, наиболее важным является то, что алгоритм метода Ньютона очень прост (мы считаем, что задача линейного программирования решается стандартной программой) и не включает в себя таких тонких и не вполне алгоритмически однозначных средств, как операция растяжения пространства.

§ 47. Линейное программирование. Симплекс-метод

Задача линейного программирования, имеющая многочисленные приложения в экономике, представляет для нас интерес как характерная промежуточная задача, возникающая в алгоритмах поиска минимума. Она формулируется обычно следующим образом: найти числа s_n , $n = 1, 2, \dots, N$ из условий

$$\min_s \sum_{n=1}^N s_n h_n^0, \quad (1)$$

$$X^i + \sum_{n=1}^{\leq N} s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \quad (2)$$

$$s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N. \quad (3)$$

Здесь h_n^i , X^i , s_n^- , s_n^+ — суть заданные числа. Введем характерные геометрические объекты, связанные с этой задачей. Прежде всего, мы имеем N -мерное пространство точек $s = \{s_1, s_2, \dots, s_N\}$ и $(m+1)$ -мерное пространство. В последнем определим векторы $X = \{0, X^1, \dots, X^m\}$, $h_n = \{h_n^0, h_n^1, \dots, h_n^m\}$, $n=1, \dots, N$. Множество допустимых по условиям (3) точек s образует прямоугольник σ в N -мерном пространстве. Рассмотрим его отображение в $(m+1)$ -мерное пространство точек $x = \{x^0, x^1, \dots, x^m\}$:

$$x = X + \sum_{n=1}^N s_n h_n. \quad (4)$$

Образом σ является выпуклый многогранник P , и задача линейного программирования (1)–(3) может быть сформулирована так:

Задача А. Пусть задан вектор $e = \{1, 0, \dots, 0\}$ в $(m+1)$ -мерном пространстве. Найти в P точку $x = \lambda e$ с наименьшим значением λ , и ее прообраз $s \in \sigma$. Другими словами, нужно найти $s \in \sigma$ таким образом, что $X + \sum s_n h_n = \lambda e$ и λ — наименьшее возможное число. В дальнейшем мы будем считать e произвольным заданным вектором; это позволит сразу же сформулировать симплекс-метод для некоторых обобщений стандартной постановки задачи. Нам будет полезно понятие «границ» P . Выделим среди чисел $1 \div N$ подмножество из m индексов $\{n_1, n_2, \dots, n_m\}$. Это подмножество обозначим M . Рассмотрим множество точек

$$x = X + \sum_{n \notin M} s_n h_n + \sum_{n \in M} s_n h_n, \quad (5)$$

причем числа s_n , $n \notin M$ закреплены в крайних положениях s_n^- или s_n^+ , а числа s_n , $n \in M$ могут изменяться на отрезках $[s_n^-, s_n^+]$. Множество таких точек x образует m -мерный многогранник. Именно из таких многогранников, соответствующих всевозможным наборам M , и состоит граница P . При различных комбинациях $s_n = [s_n^-, s_n^+]$ или s_n^+ , $n \notin M$ данная «грань» занимает положение внутри P . Однако нетрудно найти и те значения s_n , $n \notin M$, когда грань оказывается на границе P и является его гранью в обычном смысле слова. Для этого нужно найти вектор g , ортогональный всем h_n , $n \in M$, и определить граничную точку P условием

$$\min_{x \in P} (x, g) = \min_{s^- \leq s \leq s^+} (X + \sum_n s_n h_n, g). \quad (6)$$

Задача (6) решается просто:

$$s_n \begin{cases} = s_n^- & \text{при } (h_n, g) > 0, \\ = s_n^+ & \text{при } (h_n, g) < 0, \\ \in [s_n^-, s_n^+] & \text{при } (h_n, g) = 0. \end{cases} \quad (7)$$

Перейдя к вектору — g , получим вторую точную грань P , соответствующую данному набору индексов M . В настоящее время разработано большое число алгоритмов точного решения задачи A . Все они объединяются общим термином *симплекс-метод*, однако различают прямые и двойственные варианты симплекс-метода. С этими двумя вариантами связаны две основные качественные идеи, в той или иной мере лежащие в основании большинства алгоритмов как точного, так и приближенного решения задачи линейного программирования.

П р я м о й с и м п л е к с - м е т о д. Изложение этого алгоритма будет проведено по следующему плану. Сначала качественно разъясняется основная идея метода. Затем эта идея получает четкое математическое оформление. И, наконец, приводятся расчетные формулы. Последний вопрос практически важен, он связан со стремлением свести к минимуму необходимые вычисления. Алгоритм представляет собой процедуру последовательного улучшения так называемых допустимых решений (планов). *Допустимым решением* называется точка $s \in \sigma$, для которой

$$X + \sum_{n=1}^N s_n h_n = \lambda e, \quad (8)$$

однако λ может быть и не минимальным. Один стандартный шаг алгоритма состоит в переходе к следующей точке $s \in \sigma$, также образующей допустимое решение, но уже с меньшим значением λ . Алгоритм является конечным в том смысле, что через конечное число стандартных шагов создается ситуация, в которой получить новое лучшее допустимое решение нельзя, и тогда это есть решение задачи. Однако хорошей теоретической оценки числа шагов нет; практика показывает, что для решения нужно $\sim N$ шагов. Обычно под допустимым решением понимают не всякую точку $s \in \sigma$, удовлетворяющую (8), а некоторый специальный подкласс таких точек. А именно, выделяется множество M из m индексов: $M \in \{1, 2, \dots, N\}$, и в (8) предполагается

- 1) $s_n = \{s_n^- \text{ или } s_n^+\}$ при $n \notin M$;
- 2) $s_n \in [s_n^-, s_n^+]$ при $n \in M$.

Мы будем придерживаться, ради простоты, этого же правила, хотя, как нетрудно будет убедиться, это не обязательно. В целях наглядности дальнейшего изложения будет полезно ввести следующие обозначения и термины. Векторы h_n , $n \in M$ называются *базисными*. Мы обозначим их H_1, H_2, \dots, H_m и присоединим к ним еще вектор $H_0 = e$. Переменные s_n , $n \in M$ также называются *базисными*, и мы используем для них обозначения $\xi_1 = s_n, \dots, \xi_m = s_{n_m}$. Остальные переменные называются *внебазисными*, и

для них используются первоначальные обозначения. Итак, имеется допустимое решение, для которого справедливо равенство

$$X + \sum_{n \notin M} s_n h_n + \sum_{k=1}^m \xi_k H_k = \lambda e. \quad (10)$$

Вводя и для λ обозначение $\lambda = -\xi_0$, перепишем (10) в виде

$$X + \sum_{n \notin M} s_n h_n + \sum_{k=0}^m \xi_k H_k = 0. \quad (11)$$

Теперь сделаем над (11) тождественное преобразование, выбрав некоторый из внебазисных индексов $j \notin M$:

$$\left\{ X + \sum_{n \notin M} s_n h_n + \delta s_j h_j \right\} + \left\{ \sum_{k=0}^m \xi_k H_k - \delta s_j h_j \right\} = 0. \quad (12)$$

При этом ограничим изменение δs_j таким образом, чтобы было выполнено условие $s_j^- \leq s_j + \delta s_j \leq s_j^+$, т. е. первую скобку (12) можно было бы записать в виде $X + \sum s_n h_n$ с новым допустимым значением s_j , а вторую скобку также попытаемся записать в прежнем виде $\sum_k \xi_k H_k$, но уже с новыми значениями ξ_k . При этом следует стремиться к тому, чтобы новое значение ξ_0 стало больше. Для выполнения этой программы нужно уметь разлагать любой вектор h_j по базису $\{H_0, H_1, \dots, H_m\}$. Проще всего это сделать, имея биортогональный базис $\{\psi_0, \psi_1, \dots, \psi_m\} : (H_i, \psi_j) = \delta_{ij}$ (δ_{ij} — символ Кронекера). Тогда

$$h_j = \sum_{i=0}^m (h_j, \psi_i) H_i,$$

и вторая скобка в (12) преобразуется к виду

$$\sum_{k=0}^m \xi_k H_k - \delta s_j h_j = \sum_{k=0}^m [\xi_k - \delta s_j (h_j, \psi_k)] H_k. \quad (13)$$

Проверим, достигается ли увеличение ξ_0 (уменьшение λ). Если в исследуемом допустимом решении $s_j = s_j^- (s_j^+)$, то возможно лишь $\delta s_j \geq 0$ ($\delta s_j \leq 0$). Следовательно, операция приводит к успеху лишь в случае $(h_j, \psi_0) < 0$ (> 0). Таким образом, нужно просмотреть все внебазисные переменные s_n , $n \notin M$, и если ни одно из них не может быть проварьировано с уменьшением λ , задача решена. Если же в процессе испытания обнаружится индекс $j \notin M$, для которого условие убывания λ выполнено, следует продолжить вычисления и выяснить, какова наибольшая возможная вариация δs_j (ведь убывание λ пропорционально δs_j). Ограничение на δs_j связано с тем,

что новые значения $\xi'_k = \xi_k - \delta s_j (h_j, \psi_k)$ не должны выходить за рамки отрезков $[\xi_k^-, \xi_k^+]$. Таким образом, смещение δs_j ограничено некоторой величиной $|\delta s_j| \leq |\Delta|$. Могут представиться два случая: в первом ограничение на δs_j связано с условием $s_j^- \leq s_j + \delta s_j \leq s_j^+$, и в этом случае происходит только изменение внебазисной переменной: s_j переходит из положения s_j^- в s_j^+ (или наоборот), и затем такой же анализ производится с остальными внебазисными переменными. Более сложен второй случай, когда ограничение δs_j связано с тем, что одна из базисных переменных ξ_r достигает границы (ξ_r^- или ξ_r^+) при $\delta s_j = \Delta$: $\xi_r - \Delta (h_j, \psi_k) = \xi_r^-$ (или ξ_r^+), тогда как $s_j^- < s_j + \Delta < s_j^+$. При этом происходит преобразование базиса и множества M : из M выводится индекс n_r , соответствующая ему переменная становится внебазисной. И наоборот, индекс j вводится в M (под номером n_r), в базисе $\{H_0, H_1, \dots, H_m\}$ вектор H_r заменяется на h_j . Наиболее сложные преобразования связаны с необходимостью соответствующего изменения биортогональной системы. Обозначим через $\{\psi'_0, \psi'_1, \dots, \psi'_m\}$ систему, биортогональную к новому базису $\{H'_0, H'_1, \dots, H'_m\}$, и получим формулы для вычисления ψ' через ψ .

Пусть $\psi'_i = \sum_{q=0}^m \gamma_{iq} \psi_q$. Для определения коэффициентов разложения γ_{li}

используем соотношения биортогональности:

1) при $i \neq r$, $l = 0, 1, 2, \dots, m$:

$$\delta_{li} = (\psi'_i, H'_i) = \left(\sum_q \gamma_{iq} \psi_q, H_i \right) = \sum_q \gamma_{iq} (\psi_q, H_i) = \sum_q \gamma_{iq} \delta_{qi} = \gamma_{li};$$

2) при $i = r$, $l = 0, 1, \dots, m$ положим

$$\begin{aligned} H'_r &= \sum_{n=0}^m c_n H_n, \quad c_n = (h_j, \psi_n), \\ \delta_{lr} &= (\psi'_l, H'_r) = \left(\sum_q \gamma_{lq} \psi_q, \sum_n c_n H_n \right) = \\ &= \sum_q \delta_{lq} c_q = \begin{cases} c_l + \gamma_{lr} c_r & \text{при } l \neq r, \\ c_r \gamma_{rr} & \text{при } l = r. \end{cases} \end{aligned}$$

Итак, получены формулы для γ_{li} :

$$\begin{cases} \gamma_{li} = \delta_{li} & \text{при } i \neq r, \\ \gamma_{lr} = -c_l/c_r & \text{при } l \neq r, \\ \gamma_{rr} = 1/c_r. \end{cases}$$

Отсюда следуют простые формулы пересчета ψ :

$$\begin{cases} \psi'_l = \psi_l - \frac{c_l}{c_r} \psi_r, & l \neq r, \\ \psi'_r = \frac{1}{c_r} \psi_r. \end{cases}$$

Этим и заканчивается стандартный шаг симплекс-метода. Доказывать теорему о его сходимости мы здесь не будем, однако укажем на основные факторы этого доказательства.

1. Каждый шаг описанного выше алгоритма, если он осуществим, сопровождается получением нового допустимого решения с меньшим, чем было до этого, значением λ .

2. Задача носит существенно дискретный характер: для каждой неоптимальной ситуации, определяемой разбиением переменных $\{s_n\}_1^N$ на две группы — базисные и небазисные, существует свое минимальное ненулевое значение понижения λ . Этих ситуаций конечное число, следовательно, существует такое $\varepsilon > 0$, что каждый шаг симплекс-метода, в какой бы ситуации он не осуществлялся, приводит к понижению λ не менее, чем на ε . Поэтому число шагов не может быть бесконечным (предполагается, разумеется, что решение задачи существует и $\min \lambda > -\infty$, даже если среди чисел s^- , s^+ есть и бесконечные, как в задачах с односторонними ограничениями типа $0 \leq s_n$).

3. Если среди небазисных переменных нет ни одной, которую можно проварьировать с убыванием λ , это свидетельствует о том, что данное допустимое решение — оптимально. Этот факт нам будет удобно доказать несколько ниже, в связи с двойственным вариантом симплекс-метода.

Заметим, что мы не рассматриваем некоторых осложнений в связи с возможной в принципе вырожденностью задачи. Эта вырожденность, редко встречающаяся в практике экономических расчетов, может привести, например, к тому, что ограничение на δs_j оказывается связанным одновременно с выходом $s_j + \delta s_j$ и какой-то из базисных переменных $\xi_k + (\psi_k, h_j) \delta s_j$ на границу допустимого по условиям задачи отрезка $[\xi_k^-, \xi_k^+]$. В этом случае нужно разбираться, следует ли преобразовывать базис или нет. Заметим, что осложнения, связанные с некоторыми вырожденными ситуациями, принципиальных трудностей не содержат, и в соответствующей литературе описаны необходимые дополнения к алгоритмам.

Теперь приведем сводку основных расчетных формул данного варианта симплекс-метода. Расчет начинается в ситуации, когда известно некоторое допустимое решение, т. е. выделено множество базисных индексов

$$M: \{n_1, n_2, \dots, n_m\},$$

известны значения внебазисных переменных

$$s_n = \{s_n^- \text{ или } s_n^+\}, \quad n \notin M,$$

и базисных переменных

$$s_n^- \leq s_n \leq s_n^+, \quad n \in M,$$

так что выполнено соотношение

$$X + \sum_{n=1}^N s_n h_n = \lambda e.$$

Кроме того, известен биортогональный базис

$$\{\psi_0, \psi_1, \dots, \psi_m\}.$$

В тех задачах, когда нахождение такой стартовой ситуации не является очевидным, может быть использован метод введения искусственного базиса: он будет изложен несколько позже.

I. Для всех $n \notin M$ определяются

$$\alpha = \langle h_n, \psi_0 \rangle.$$

Если $\{\alpha < 0 \text{ и } s_n = s_n^+\}$ или $\{\alpha \geq 0 \text{ и } s_n = s_n^-\}$, никаких операций не делается, и переходим к аналогичным вычислениям для следующего $n \notin M$. Если весь список $n \notin M$ исчерпан и встречались только такие ситуации, задача решена.

II. Пусть встрети́лась ситуация

$$\{\alpha > 0 \text{ и } s_n = s_n^+\} \quad \text{или} \quad \{\alpha < 0 \text{ и } s_n = s_n^-\}.$$

Тогда вычисляются:

$$\text{при } \alpha > 0: \quad \Delta^* = s_n^- - s_n^+ \quad (\text{т. е. } \delta s_n < 0),$$

$$\text{при } \alpha < 0: \quad \Delta^* = s_n^+ - s_n^- \quad (\text{т. е. } \delta s_n > 0).$$

III. Для $k = 1, 2, \dots, m$ вычисляются: $c_k = \langle h_n, \psi_k \rangle$

$$\text{при } \alpha c_k > 0 \quad (\text{т. е. } \delta \xi_k > 0): \quad \Delta_k = \frac{\xi_k^+ - \xi_k}{c_k},$$

$$\text{при } \alpha c_k < 0 \quad (\text{т. е. } \delta \xi_k < 0): \quad \Delta_k = \frac{\xi_k^- - \xi_k}{c_k}.$$

IV. Находятся

$$\Delta = \min_{k=1, \dots, m} |\Delta_k| \quad \text{и} \quad r: \Delta = |\Delta_r|.$$

V. Если $|\Delta^*| < |\Delta|$, пересчитывается внебазисная переменная

$$s_n := s_n + \Delta$$

и пересчитываются все базисные переменные ξ_k (т. е. s_{nk}): $\xi_k := \xi_k - \Delta c_k$, $k = 1, 2, \dots, m$, и переходим к I.

VI. Если $|\Delta^*| > |\Delta|$, происходит изменение переменных и базиса:

- а) вычисляется $\delta s = -|\Delta| \operatorname{sign} \alpha$, и $s_n := s_n + \delta s$,
- б) вычисляются новые значения базисных переменных

$$s_{n_k} := s_{n_k} - \delta s c_k, \quad k = 1, 2, \dots, m,$$

- в) индексу n_r присваивается значение n ,
- г) пересчитывается базис

$$\psi_l := \psi_l - \frac{c_l}{c_r} \psi_r, \quad l \neq r,$$

$$\psi_r := \frac{1}{c_r} \psi_r.$$

Вычисления далее продолжаются с пункта I.

Число операций, необходимое для реализации одного шага алгоритма, может быть без труда оценено. Наиболее трудоемкими являются блоки I, требующий вычисления $(N-m)$ скалярных произведений, т. е. $\sim(N-m)t$ операций, затем III, требующий вычисления t скалярных произведений ($\sim t^2$ операций), и, наконец, пересчет ψ (VI) требует $\sim t^2$ операций. Остальные вычисления не существенны, так как необходимое для них число операций растет линейно с ростом N и t . Общее число операций $Q = C_1 Nm + C_2 t^2$, причем коэффициенты C_1 и C_2 невелики. Заметим, что имея резерв памяти, можно сократить число операций. В самом деле, пусть величины $\alpha_n = (h_n, \psi_0)$ запоминаются. Тогда после пересчета базиса ψ легко пересчитать и α :

$$\alpha'_n = (\psi'_0, h_n) = \left(\psi_0 - \frac{c_0}{c_r} \psi_r, h_n \right) = \alpha_n - c_0,$$

и для вектора h_{n_r} , выведенного из базиса,

$$\alpha'_{n_r} = (h_{n_r}, \psi'_0) = \left(h_{n_r}, \psi_0 - \frac{c_0}{c_r} \psi_r \right) = -\frac{c_0}{c_r}.$$

При этом объем вычислений сокращается до St^2 . Наконец, заметим, что в больших задачах экономического содержания, когда $N, t \sim 10^2 - 10^3$, как правило, матрица исходной формулировки очень слабо заполнена: подавляющая часть ее элементов — нули. Вводится характеристика заполненности матрицы μ , равная отношению числа ненулевых элементов к Nt . При соответствующей организации программы в памяти хранятся только ненулевые элементы h , и вычисления производятся только с ними. Поэтому вычисление скалярных произведений типа (h, ψ) требует уже не t операций, а μt . Векторы же ψ в общем случае имеют все ненулевые элементы. Поэтому объем вычислений при пересчете базиса не зависит от μ . С учетом этих соображений, коли-

чество операций на итерацию сводится к $Q = C_1 \mu m^2 + C_2 m^3$. Как уже отмечалось, опыт расчетов показал, что в среднем число итераций в симплекс-методе $\sim N$, поэтому трудоемкость решения задачи линейного программирования оценивается по порядку величины числом Nm^2 . Другой важной характеристикой является необходимый ресурс памяти. В основном требования к памяти определяются необходимостью запоминать матрицу h (это μNm ячеек) и базис ψ (m^2 ячеек). Остальные объекты — линейные. Заметим, что выше был описан вариант алгоритма, в котором исходная матрица задачи h не преобразуется в процессе решения, но появляется новая матрица $\{\psi\}$. Есть и другой вариант, в котором в процессе вычислений все векторы h представляются коэффициентами разложения по базису. В этом случае векторы базиса $\{H_0, H_1, \dots, H_m\}$ образуют единичную матрицу, и нет необходимости в базисе $\{\psi\}$, однако текущее состояние матрицы h , вообще говоря, полностью заполняет таблицу Nm , и первоначальная слабая заполненность матрицы задачи не используется.

Задача линейного программирования с неравенствами. Часто задача линейного программирования формулируется в таком виде: найти s_n из условий

$$\text{I.} \quad \min_s \sum_{n=1}^N s_n h_n^0,$$

$$\text{II.} \quad X^i + \sum_{n=1}^N s_n h_n^i \leq 0, \quad i = 1, 2, \dots, m,$$

$$\text{III.} \quad 0 \leq s_n, \quad n = 1, 2, \dots, N.$$

Прежде всего, удобно условия II привести к стандартной форме, введя еще дополнительные векторы $h_{N+1} = \{0, \dots, 0, 1_i, 0, \dots, 0\}$ *) и записав II в виде

$$X^i + \sum_{n=1}^{N+m} s_n h_n^i = 0, \quad i = 1, 2, \dots, m,$$

$$s_n \geq 0, \quad n = 1, 2, \dots, N+m.$$

Что касается односторонних ограничений $s_n \geq 0$, то они остаются без изменений, и это приводит лишь к упрощению алгоритма: в процессе итераций (если решение существует и λ ограничено снизу) нужно учитывать только ограничение на переменные s_n снизу. В таких задачах обычно не составляет труда выбор начального базиса $\{H\}$ и биортогонального к нему $\{\psi\}$. В общем случае используется

*) Здесь и в дальнейшем 1_i означает 1 на i -м месте.

Метод введения искусственного базиса. Всем переменным задачи s_n присваивается, например, значение s_n^- , но исходная формулировка искажается введением m дополнительных векторов и переменных. Именно эти дополнительные векторы и образуют первоначальный базис. Например, если $e = \{1, 0, 0, \dots, 0\}$, то базис $\{H\}$ и биортогональный к нему $\{\psi\}$ имеют вид

H_0^0	H_1	H_2	\dots	H_m
1	c_1	c_2	\dots	c_m
0	1	0	\dots	0
0	0	1	\dots	0
\dots	\dots	\dots	\dots	\dots
0	0	0	\dots	1

ψ_0	ψ_1	ψ_2	\dots	ψ_m
1	0	0	\dots	0
$-c_1$	1	0	\dots	0
$-c_2$	0	1	\dots	0
\dots	\dots	\dots	\dots	\dots
$-c_m$	0	0	\dots	1

Начальные значения базисных переменных s_{N+i} , $i=1, 2, \dots, m$ выбираются из соотношений

$$X^i + \sum_{n=1}^N s_n^- h_n^i + s_{N+i} = 0.$$

Выбор s_{N+i}^+ , C_i следует осуществить так, чтобы решение новой, расширенной задачи совпало с решением исходной. Пусть $s_{N+i} > 0 (< 0)$. Тогда положим $s_{N+i}^- = 0$, $s_{N+i}^+ = 2s_{N+i}$ ($s_{N+i}^- = 2s_{N+i}$, $s_{N+i}^+ = 0$) и в качестве C_i возьмем очень большое по абсолютной величине положительное (отрицательное) число, с тем, чтобы минимальность λ в расширенной задаче была несовместима с $s_{N+i} > 0 (< 0)$. Этими соображениями мы и ограничимся.

Двойственный симплекс-метод. Начнем с анализа геометрической картины, связанной с задачей линейного программирования. На рис. 76 изображен (качественно) многогранник P , причем одна ось — прямая le , в качестве второй «оси» на рис. 76 принято m -мерное пространство. Граница P состоит из m -мерных граней (на рис. 76 они изображены отрезками). Каждая грань определяется вектором g , ортогональным данной грани. Мы будем считать этот вектор нормированным условием $(g, e) = 1$. Такие векторы определяют нижние грани P , при нормировке $(g, e) = -1$ получим верхние. Это следует понимать так: коль скоро задан вектор g , соответствующая ему грань определяется как совокупность точек x вида

$$x = X + \sum_{n=1}^N s_n h_n, \quad (14)$$

причем

$$s_n \begin{cases} = s_n^- & \text{при } (h_n, g) > 0, \\ = s_n^+ & \text{при } (h_n, g) < 0, \\ \in [s_n^-, s_n^+] & \text{при } (h_n, g) = 0. \end{cases}$$

Заметим сразу же, что произвольному вектору g обычно соответствует вершина, т. е. нет ни одного индекса n , для которого $(h_n, g) = 0$. Однако при реализации двойственного симплекс-метода

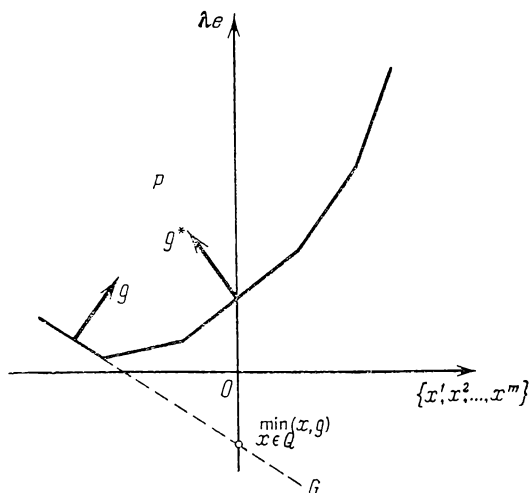


Рис. 76.

типичной является ситуация, когда таких векторов ровно m ; множество таких индексов, как и выше, обозначим $M = \{n_1, n_2, \dots, n_m\}$, не исключая, однако, возможности, что M содержит и меньше, чем m индексов. Но мы предположим, что задача является невырожденной, т. е. любая группа из $(m+1)$ -векторов $\{H_0, H_1, \dots, H_m\}$, $H_0 = e$, $H_k = h_{n_k}$ линейно-независима. Таким образом, множество M не может содержать более m индексов. Точка x (14) является решением задачи $F(g) = \min_{x \in P} (x, g)$, причем величина $F(g)$ имеет простой геометрический смысл: если провести через x (14) гиперплоскость G , ортогональную g , то она пересечет прямую λe в точке $F(g)e$. Это следует из уравнения для точки пересечения λe : $(\lambda e - x, g) = 0$ и условия нормировки $(e, g) = 1$. Разумеется, в качестве x можно брать любую из точек грани, определяемой формулой (14). Среди образующих границу P граней существует (если существует решение задачи, т. е. прямая λe пересекает P) и такая, определяемая вектором g^* , которая пересекается

с прямой λe в искомой точке Λe ($\Lambda = \min \lambda$). Эта грань M^* определяется уравнением $\max_g \min_{x \in P} (x, g)$. Оно является следствием очевидных соотношений

$$F(g) = \min_{x \in P} (x, g) \leq (\Lambda e, g) = \Lambda$$

и $F(g^*) = \Lambda$.

Двойственный симплекс-метод основан на том, что, исходя из некоторого вектора g , устраивают последовательные его повороты, причем каждый поворот (переход от g к g') сопровождается ростом $F(g)$. Этим обеспечивается сходимость $g \rightarrow g^*$; признаком того, что текущий вектор g уже есть g^* , служит невозможность его изменения. Однако, после того как встретила *неулучшаемая* грань M^* , следует еще решить систему $(m+1)$ линейных уравнений

$$X + \sum_{n \notin M^*} s_n h_n + \sum_{n \in M^*} \xi_n h_n - \xi_0 e = 0, \quad (15)$$

в которой s_n , $n \notin M^*$ фиксированы в соответствии с (14), ξ_n — искомые неизвестные, которые автоматически окажутся ограниченными заданными условиями $s_n^- \leq \xi_n \leq s_n^+$, а переменная ξ_0 будет равна $F(g^*) = \Lambda$. Впрочем, последняя величина уже найдена и число неизвестных в (15) можно сократить на единицу. Теперь опишем алгоритм поворота. Итак, пусть имеется некоторая стандартная ситуация: выделено текущее множество базисных индексов M , имеются базис $\{H_0, H_1, \dots, H_m\}$ и биортогональный к нему $\{\psi_0, \psi_1, \dots, \psi_m\}$, причем $H_0 = e$, а ψ_0 есть стандартизованное обозначение вектора g . Исследуемая грань есть m -мерный выпуклый многогранник, его границу образуют $2m$ $(m-1)$ -мерных ребер. Они определяются тем, что одно из переменных s_{n_k} фиксируется в положении s^- (или s^+), остальные базисные переменные могут меняться в интервалах $[s^-, s^+]$. Теперь фиксируем один из базисных индексов r и попытаемся «повернуть» вектор g (ψ_0) вокруг соответствующего «ребра». Это означает, что рассматривается новый вектор $\psi'_0 = \psi_0 + \alpha \psi_r$, где α — неопределенный пока скаляр. Очевидно, что при любом α имеем $(\psi'_0, e) = (\psi_0 + \alpha \psi_r, H_0) = 1$ и $(\psi'_0, H_k) = (\psi_0 + \alpha \psi_r, H_k) = 0$ при $k \neq r$ (последнее, собственно говоря, и означает, что рассматривается поворот вокруг «ребра», соответствующего вектору H_r). Теперь следует определить параметр α с наибольшим выигрышем в величине $F(g)$. Таким образом, для определения α имеем задачу

$$\max_{\alpha} F(\psi_0 + \alpha \psi_r) = \max_{\alpha} \min_{x \in P} (x, \psi_0 + \alpha \psi_r). \quad (16)$$

Однако сначала, не решая задачу (16), нужно выяснить, можно ли вообще поворотом вокруг данного ребра получить рост $F(g)$. Это зависит от производной $F(\psi_0 + \alpha \psi_r)$ по α в точке $\alpha = 0$. Нужно

только иметь в виду, что $F(\psi_0 + \alpha \psi_r)$, вообще говоря, производной не имеет: она имеет лишь производные по направлениям ($\alpha > 0$ и $\alpha < 0$). Суть дела поясняет рис. 77, на котором представлены два возможных варианта. На этом рисунке изображена, так сказать, проекция P , причем вершины изображают $(m-1)$ -мерные ребра исследуемой грани.

В каждой ситуации отмечены векторы ψ_0, ψ'_0 — соответствующий, например, малому положительному α , и ψ'' — малому отрицательному. Видно, что переход к ψ' сопровождается (в ситуации а))

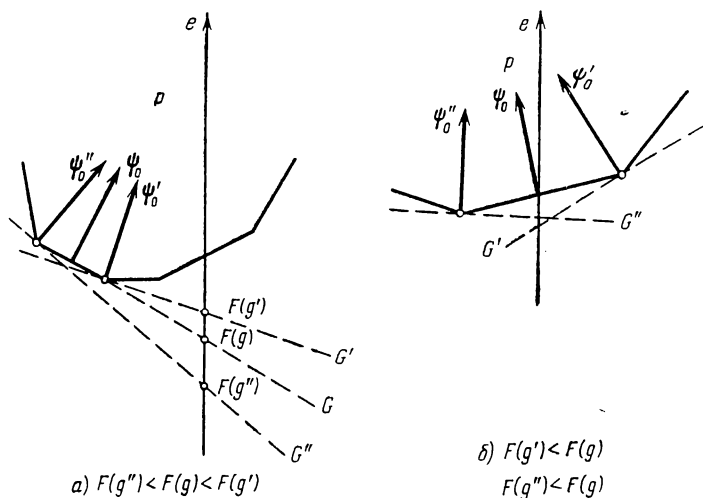


Рис. 77.

ростом $F(g)$, а переход к ψ'' — падением. В ситуации б) переход как к ψ' , так и к ψ'' приводит к падению $F(g)$. Теперь следует оформить эти качественные соображения аналитически. Мы используем то, что $\min_{x \in P} (x, \psi_0 + \alpha \psi_r)$ при достаточно малых α достигается только в точках выделенных двух «ребер» исследуемой грани. Это следует из непрерывности по α величин $(h_n, \psi_0 + \alpha \psi_r)$ и из предположения о невырожденности задачи — т. е. из того, что для всех $n \notin M$ $|(h_n, \psi_0)| \geq \varepsilon > 0$. Таким образом, при малых α в (14) может измениться только одна базисная переменная ξ_r , которая станет либо ξ_r^- , либо ξ_r^+ ; для остальных $\xi_k, k \neq r$ по-прежнему $(\psi_0 + \alpha \psi_r, H_k) = 0$. Обозначим через x^- точку x , принадлежащую исследуемой грани, в которой базисная переменная $\xi_r = \xi_r^-$, x^+ пусть обозначает точку грани, полученную по (14) при $\xi_r = \xi_r^+$; значения остальных базисных переменных безразличны. Для определенности, пусть $\xi_r = \xi_r^+$. Очевидно, $x^+ = x^- + (\xi_r^+ - \xi_r^-) H_r$.

Теперь можно для достаточно малых α найти

$$\min_{x \in P} (x, \psi_0 + \alpha \psi_r) = \min \{(x^-, \psi_0 + \alpha \psi_r), (x^+, \psi_0 + \alpha \psi_r)\}.$$

Далее:

$$\begin{aligned} (x^-, \psi_0 + \alpha \psi_r) &= (x^-, \psi_0) + \alpha (x^-, \psi_r) = F(g) + \alpha (x^-, \psi_r), \\ (x^+, \psi_0 + \alpha \psi_r) &= (x^+, \psi_0) + \alpha (x^+ + (\xi_r^+ - \xi_r^-) H_r, \psi_r) = \\ &= F(g) + \alpha (x^-, \psi_r) + \alpha (\xi_r^+ - \xi_r^-). \end{aligned}$$

Итак, для малых α

$$\min_{x \in P} (x, \psi_0 + \alpha \psi_r) = \begin{cases} F(g) + \alpha (x^-, \psi_r) & \text{при } \alpha \geq 0, \\ F(g) + \alpha [(x^-, \psi_r) + (\xi_r^+ - \xi_r^-)] & \text{при } \alpha \leq 0. \end{cases} \quad (17)$$

Таким образом, поворот вокруг данного «ребра», имеет смысл в случае, когда либо $(x^-, \psi_r) > 0$, либо $(x^-, \psi_r) + (\xi_r^+ - \xi_r^-) < 0$ (это признак ситуации типа *a* на рис. 77). Если же $(x^-, \psi_r) < 0$ и $(x^-, \psi_r) + (\xi_r^+ - \xi_r^-) > 0$, то поворот может привести лишь к убытыванию $F(g)$. Это есть признак ситуации типа *b*) рис. 77; не следует только трактовать его как признак пересечения прямой l_e с исследуемой гранью: ведь на рис. 77 изображена проекция. Итак, имея в качестве исходной информации базиса $\{H\}$, $\{\psi\}$ и точку x^- , следует для всех $n \in M$ (или для $k=1, 2, \dots, m$) вычислить (x^-, ψ_r) и проверить условие

$$(x^-, \psi_r) > 0 \quad \text{или} \quad [(x^-, \psi_r) + (\xi_r^+ - \xi_r^-)] < 0. \quad (18)$$

Как только встретится индекс r , для которого имеет место (18), осуществляется поворот. Он требует определения параметра α . Здесь возможны два варианта — локальный и глобальный. Однако прежде заметим, что невыполнение (18) свидетельствовало о невыгодности поворота вокруг r -го ребра для малых α . Но в силу того, что, например,

$$\min_{x \in P} (x, \psi_0 + \alpha \psi_r) \leq (x^-, \psi_0 + \alpha \psi_r),$$

тот же вывод справедлив и для конечных α . Итак, для определения α можно решить задачу (16); это глобальный подход. Он реализован в итерационном алгоритме, изложение которого будет дано в следующем параграфе. Это более эффективный, но и более трудоемкий способ, чем локальный выбор α . Именно последний и будет здесь описан. Разница между глобальным и локальным выбором α может быть пояснена рис. 77, *a*). В первом случае поворот приведет к новой грани, которая в проекции пересекает ось l_e , во втором — к грани, смежной с исследуемой. Следующие ниже операции должны выяснить, какой из внебазисных векторов h_n должен быть введен в базис вместо выбывающего из него вектора H_r . Искомая смежная грань есть «произведение» соответствующего H_r «ребра» и некоторого внебазисного вектора h_n . Прежде

всего следует выяснить, какое из двух соответствующих H_r «ребер» будет ребром новой грани. Это зависит от знака (x_r^-, ψ_r) : если эта величина положительна, то в (17) «работает» первая из формул правой части, $\alpha > 0$, и при выходе из базиса переменная $s_{nr} = s_{nr}^-$ ($\xi_r = \xi_r^-$). В противном случае $\alpha < 0$, $s_{nr} = s_{nr}^+$. Что касается вводимого в базис вместо H_r вектора h_n , то он (при $\alpha > 0$) определяется задачей; найти

$$\min_{n \notin M} \alpha \quad \text{при условии} \quad (h_n, \psi_0 + \alpha \psi_r) = 0.$$

Таким образом, следует, перебирая индексы $n \notin M$, вычислять

$$\alpha_n = -(h_n, \psi_0) / (h_n, \psi_r)$$

и среди положительных α_n найти наименьшее (и его индекс n). Вариант с $\alpha < 0$ аналогичен. После того как определен вводимый в базис вектор, следует пересчитать биортогональную систему. Это делается точно так же, как и в прямом симплекс-методе. Кроме того, нужно получить новую точку x^- ; она связана со старой очевидной формулой

$$x^- := x^- + (s_n^- - s_n)h_n + (\xi_r^- - \xi_r)H_r$$

(s_n, ξ_n — старые значения переменных для точки x^-). Количество операций, связанных с одним шагом этого варианта симплекс-метода, подсчитывается без труда и приводит примерно к той же оценке, что и в прямом варианте. Если для всех $n \in M$ условие (18) не выполняется, это свидетельствует о том, что (в случае невырожденной задачи) минимум найден, и для завершения процесса решения осталось найти значения базисных переменных, решив систему линейных уравнений (15). Что касается начала процесса, то его можно осуществить с помощью такого же искусственного базиса, который был выше описан.

Нестандартные задачи линейного программирования. Рассмотрим некоторые задачи, встречающиеся при решении задач с функциями, не имеющими производных, но дифференцируемыми по направлениям. В § 46 читатель может познакомиться с тем, каким образом возникают такие задачи. Здесь же будет показано, что они сводятся к стандартной задаче линейного программирования.

Задача A' . Найти числа s_n , $n = 1, 2, \dots, N$ из условий

$$\text{I.} \quad \min_s \max_{k=1, \dots, j} \left\{ X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right\},$$

$$\text{II.} \quad X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m,$$

$$\text{III.} \quad s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N$$

($X^{0,k}, X^i, h_n^{0,k}, h_n^i, s_n^-, s_n^+$ — заданные числа).

Задача A^* . Пусть σ^* — прямоугольник в $(N+j)$ -мерном пространстве точек $\{s_n\}_{n=1}^{N+j}$, определяемый неравенствами $s_n^- \leq s_n \leq s_n^+$, $n = 1, 2, \dots, N$;

$$0 \leq s_{N+k} \leq \infty, \quad k = 1, 2, \dots, j.$$

Пусть P — его образ в линейном отображении $(N+j)$ -мерного пространства $\{s\}$ в $(m+j)$ -мерное пространство точек $x = \{x^{0,1}, \dots, x^{0,j}, x^1, \dots, x^m\}$.

Это отображение задается формулами:

$$\begin{aligned} x^{0,k} &= X^{0,k} + \sum_{n=1}^{N+j} s_n h_n^{0,k}, \quad k = 1, 2, \dots, j, \\ x^i &= X^i + \sum_{n=1}^{N+j} s_n h_n^i, \quad i = 1, 2, \dots, m, \end{aligned}$$

причем все числа те же, что и в задаче A' , а векторы h_{N+k} суть орты вида $h_{N+k} = \{0, \dots, 0, 1_k, 0, \dots, 0\}$. Определим $(m+j)$ -мерный вектор $e = \{1, \dots, 1_k, 0, \dots, 0\}$ и поставим задачу отыскания в P точки le с минимальным значением λ . Задача A^* является задачей линейного программирования, как она сформулирована и исследуется выше. Стандартная формулировка задачи линейного программирования, принятая в большинстве руководств, соответствует частному случаю $e = \{1, 0, 0, \dots, 0\}$.

Т е о р е м а 2. Задачи A' и A^* эквивалентны (в том смысле, что решение одной из них непосредственно дает решение другой.)

Доказательство состоит из двух частей. Обозначим через $\{s'_n\}$ решение задачи A' , а через Λ' — минимальное значение формы $\max_k \left\{ X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right\}$. Через $\{s_n^*\} \in \sigma^*$ обозначим решение задачи A^* , а Λ^* — минимальное значение λ при $le \in P$.

1. Пусть известна точка s' . Дополним ее до точки $s \in \sigma^*$, доопределив лишь компоненты s_{N+k} , $k = 1, 2, \dots, j$. А именно:

$$s_{n+k} = \begin{cases} 0 & \text{при } X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} = \Lambda', \\ \Lambda' - \left\{ X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right\} > 0 & \text{в противном случае.} \end{cases}$$

Очевидно, что $s_{N+k} \geq 0$ (т. е. $\{s_n\} \in \sigma^*$) и, так как $\{s'_n\}$ удовлетворяет условиям II задачи A' , точка $\{s_n\}$ отображается в точку $\Lambda'e$. Таким образом, $\Lambda^* \leq \Lambda'$.

2. Докажем обратное неравенство $\Lambda' \leq \Lambda^*$, предположив, что известно решение $\{s_n^*\}$ задачи A^* . Образует точку $\{s_n\}_{n=1}^N$, взяв лишь первые N компонент s_n^* . Так как $\{s^*\}$ отображается

в Λ^*e , то условие II задачи A' выполнено. Покажем теперь, что

$$\max_k \left\{ X^{0,k} + \sum_{n=1}^N s_n^* h_n^{0,k} \right\} = \Lambda^*. \quad (19)$$

В самом деле, так как s^* отображается в Λ^*e , то для всех $k=1, 2, \dots, j$ имеем

$$x^{0,k} = X^{0,k} + \sum_{n=1}^N s_n^* h_n^{0,k} + s_{N+k}^* = \Lambda^*,$$

т. е. $X^{0,k} + \sum_{n=1}^N s_n^* h_n^{0,k} = \Lambda^* - s_{N+k}^* \leq \Lambda^*$, $k=1, \dots, j$. По крайней мере одна из величин s_{N+k}^* , $k=1, 2, \dots, j$ равна нулю. Если бы для всех $k=1, 2, \dots, j$ было $s_{N+k}^* \geq \Delta > 0$, то точка s^* не была бы решением задачи A' ; в этом случае, заменив в $\{s_n^*\}_{n=1}^{N+j}$ компоненты s_{N+k}^* на $s_{N+k}^* - \Delta \geq 0$, мы получили бы точку из σ^* , отображающуюся в $(\Lambda^* - \Delta)e$. Таким образом, (19) установлено и, следовательно, $\Lambda' \leq \Lambda^*$. Итак, $\Lambda' = \Lambda^*$, и доказательство закончено.

Рассмотрим следующую пару задач.

Задача В'. Найти числа $\{s_n\}_{n=1}^N$ из условий

$$\text{I.} \quad \min_s \left\{ X^0 + \sum_{n=1}^N s_n h_n^0 + \sum_{k=1}^j \left| X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right| \right\},$$

$$\text{II.} \quad X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i=1, 2, \dots, m,$$

$$\text{III.} \quad s_n^- \leq s_n \leq s_n^+ \quad n=1, 2, \dots, N$$

(или $s \in \sigma'$).

Задача В*. Пусть σ^* — прямоугольник в $(N+j)$ -мерном пространстве точек $s = \{s_1, \dots, s_N, \xi_1, \dots, \xi_j, \eta_1, \dots, \eta_j\}$ (для наглядности здесь введены особые обозначения для дополнительных переменных ξ, η : в стандартной форме их следовало бы обозначить $\xi_1 = s_{N+1}, \dots$ и т. д.). Этот прямоугольник определяется неравенствами

$$s_n^- \leq s_n \leq s_n^+, \quad n=1, \dots, N; \quad \xi_k \geq 0; \quad \eta_k \geq 0, \quad k=1, \dots, j.$$

Обозначим через P образ σ^* в линейном отображении точек s в точки x $(m+k+1)$ -мерного пространства. Это отображение задается формулами

$$x^0 = X^0 + \sum_{n=1}^N s_n h_n^0 + \sum_{k=1}^j (\xi_k + \eta_k),$$

$$x^{0,k} = X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} - \xi_k + \eta_k, \quad k=1, 2, \dots, j,$$

$$x^i = X^i + \sum_{n=1}^N s_n h_n^i, \quad i=1, 2, \dots, m.$$

Требуется найти точку $\lambda e \in P$ с наименьшим значением λ (т. е. $\min x^0$ при $x^{0,k}=0, x^i=0$).

Теорема 3. Задачи B' и B^* — эквивалентны.

Доказательство. 1. Пусть s' — решение задачи B' , Λ' — минимум формы I в B' . Сконструируем точку $s \in \sigma^*$, отображающуюся в $\Lambda' e \in P$. Этим будет доказано неравенство $\Lambda^* \leq \Lambda'$ (Λ^* — минимум λ в задаче B^*). Первые N компонент s совпадают с s' ,

нужно лишь определить ξ_k, η_k ; обозначим $a_k = X^{0,k} + \sum_{n=1}^N h_n^{0,k} s_n$ и положим

$$\begin{aligned} \xi_k &= a_k; \quad \eta_k = 0 && \text{при } a_k \geq 0, \\ \xi_k &= 0; \quad \eta_k = -a_k && \text{при } a_k < 0. \end{aligned}$$

В этом случае, очевидно, $|a_k| = \xi_k + \eta_k$, т. е.

$$x_0 = \sum_{n=1}^N s_n h_n^0 + \sum_{k=1}^j |a_k| = \Lambda' \quad \text{и} \quad a_k - \xi_k + \eta_k = 0 \quad (x^{0,k}=0),$$

т. е. s действительно отображается в $\Lambda' e$.

2. Пусть $\{s^*, \xi^*, \eta^*\}$ — решение задачи A^* . Тогда точка s^* удовлетворяет условиям II и III задачи B' . Покажем, что $X^0 + \sum_{n=1}^N s_n^* h_n^0 + \sum_{k=1}^j \left| X^{0,k} + \sum_{n=1}^N s_n^* h_n^{0,k} \right| = \Lambda^*$ и, следовательно, $\Lambda^* \geq \Lambda'$. Существенным для этого является следующий факт: в решении задачи B^* для каждого $k=1, 2, \dots, j$ лишь одно из чисел ξ_k^* или η_k^* может быть отлично от нуля. Это есть следствие следующей простой леммы.

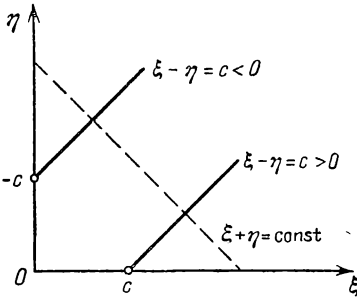


Рис 78.

Лемма 1. Минимум формы $\xi + \eta$ на линии $\xi - \eta = c$ при $\xi \geq 0, \eta \geq 0$ достигается либо в точке $\xi = c; \eta = 0$ (при $c \geq 0$), либо в точке $\xi = 0; \eta = -c$ (при $c < 0$).

Доказательство леммы немедленно следует из рис. 78, на котором изображены линия $\xi + \eta = \text{const}$ и линии $\xi - \eta = c$. Так как в решении B^* $x^{0,k} = a_k - \xi_k^* + \eta_k^*$, то $\xi_k^* = a_k, \eta_k^* = 0$ при $a_k \geq 0$. Следовательно, $\xi_k^* + \eta_k^* = |a_k|$, и доказательство завершено.

Задачи A', B' возникают при решении методом Ньютона следующих задач условной минимизации.

Задача А. Найти x из условий

$$\min_x \max_{k=1, \dots, j} f_k(x),$$

$$g_i(x) = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \\ x^- \leq x \leq x^+.$$

Задача В. Найти x из условий

$$\min_x \sum_{k=1}^j |f_k(x)|, \\ g_i(x) = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \\ x^- \leq x \leq x^+.$$

Термин «метод Ньютона» связан с тем, что для решения этих задач используется фундаментальная для вычислительной математики конструкция: нелинейная задача линеаризуется в окрестности некоторой точки x , и решением возникшей линеаризованной задачи определяются вариация аргумента δx и переход к следующему приближению $x + \delta x$. Заметим лишь, что ограничения типа $s^- \leq s \leq s^+$ обеспечивают не только выполнение исходных ограничений $x^- \leq x \leq x^+$, но достаточную малость δx , требуемую возможностью использовать линейную аппроксимацию задачи. Встречается в приложениях.

Задача С. Найти x из условий

$$\min_x \max_{k=1, \dots, j} |f_k(x)|, \\ g_i(x) = 0 \quad (\leq 0), \quad i = 1, 2, \dots, m, \\ x^- \leq x \leq x^+.$$

Линеаризуя ее, получаем задачу следующего типа:

Задача С'. Найти s_n , $n=1, \dots, N$, из условий

$$\min_s \max_{k=1, \dots, j} \left| X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right|, \\ X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \\ s_n^- \leq s_n \leq s_n^+, \quad n = 1, 2, \dots, N.$$

Она сводится к следующей задаче линейного программирования.

Задача С*. Пусть σ^* — прямоугольник в $(N+2j)$ -мерном пространстве точек $\{s_n, \xi_k, \eta_k\}$, $n=1, \dots, N$; $k=1, \dots, j$, определяемый неравенствами:

$$s_n^- \leq s_n \leq s_n^+; \quad 0 \leq \xi_k \leq S; \quad -S \leq \eta_k \leq 0; \\ S = 2 \max_k |X^{0,k}|.$$

Пусть P — образ в линейном отображении $\{s, \xi, \eta\} \in \sigma^*$ в $(m+2j)$ -мерное пространство, определяемом формулами

$$\begin{aligned}x^i &= X^i + \sum_{n=1}^N s_n h_n^i, \quad i = 1, 2, \dots, m, \\x^{m+k} &= X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} + \xi_k, \quad k = 1, 2, \dots, j, \\x^{m+j+k} &= X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} + \eta_k, \quad k = 1, 2, \dots, j.\end{aligned}$$

Найти точку $\lambda e \in P$ с минимальным значением λ и ее прообраз в σ^* . Здесь

$$e = \{\underbrace{0, 0, \dots, 0}_m, \underbrace{1, 1, \dots, 1}_j, \underbrace{-1, -1, \dots, -1}_j\}.$$

Теорема 4. Задачи C' и C^* эквивалентны.

Доказательство опускается; оно аналогично проведенным выше.

О задачах на быстрое действие. В этих задачах ищется управление $\{u(\cdot), T\}$ с целью минимизации T при выполнении условий $F_i[u(\cdot), T] = 0, i = 1, 2, \dots, m, u(t) \in U$. Обычным образом приходим к задаче линейного программирования типа:

$\min \delta T$ при условиях $X^i + \sum_{n=1}^N s_n h_n^i + \delta T h^i = 0, i = 1, \dots, m, s_n^- \leq s_n \leq s_n^+$. Таким образом, лишь один коэффициент $h_{N+1}^0 = 1$, остальные $h_n^0 = 0$. Задача оказывается существенно вырожденной, и опыт ее решения в такой форме (итерационным методом § 48) оказался неудачным: требовалось слишком большое число итераций для достижения нужной точности. Поэтому в таких задачах использовался следующий прием, приводящий, как показали вычисления, к существенно более простой и легко решаемой задаче. Запишем задачу в виде

$$\min \lambda \quad \text{при условии} \quad X + \sum_{n=1}^N s_n h_n = \lambda e,$$

где $e = -h/\|h\|$, $\lambda = \delta T\|h\|$. Это уже стандартная задача, без вырождения. Именно ее и рекомендуется использовать при решении задач быстрого действия (строка h^0 теперь отсутствует). Заметим, что при решении задачи линейного программирования типа

$$\min_s \max_{k=1, \dots, j} \left\{ X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} \right\}$$

часто рекомендуют сведение к стандартной (с $e = \{1, 0, \dots, 0\}$) введением дополнительной переменной ξ :

$$\min \xi \quad \text{при условиях} \quad X^{0,k} + \sum_{n=1}^N s_n h_n^{0,k} - \xi \leq 0, \\ k = 1, 2, \dots, j.$$

Имеющийся у автора опыт (впрочем, существенно связанный с итерационным методом § 48) заставляет осторожно относиться к этому сведению, отдавая предпочтение тому способу решения (с $e = \{1, 1, \dots, 1, 0, \dots, 0\}$), который был изложен выше.

§ 48. Линейное программирование. Итерационный метод

Хотя для решения задачи линейного программирования существуют четкие конечные методы (они описаны в § 47), не прекращается работа по созданию итерационных, приближенных методов. Для этого есть по крайней мере две причины. Дело в том, что реализация симплекс-метода встречает определенные трудности в экономических задачах высокой размерности ($N, m \sim 10^3$). В таких задачах работа с матрицей объемом 10^6 ячеек памяти становится очень сложной. В то же время исходная матрица задачи, будучи слабо заполненной, часто может быть размещена в оперативной памяти машины. Встречаются задачи, элементы матрицы которой можно вообще не запоминать, а вычислять по сравнительно простым формулам. В таких ситуациях итерационные методы, не преобразующие исходной формы задачи и не порождающие новых объектов типа матрицы общего положения (как, например, биортogonalный базис $\{\psi\}$), несмотря на значительно меньшую надежность, могут оказаться предпочтительными и даже единственно реализуемыми. Для нас же будет важна и другая причина, заставляющая обратиться к итерационным методам. Ведь задачи линейного программирования, возникающие при решении задач оптимального управления, являются конечно-разностными аппроксимациями *континуальных задач*: найти функцию $\delta u(t)$ из условий

$$\min_{\delta u(\cdot)} \int_0^T w_0(t) \delta u(t) dt, \\ F_i + \int_0^T w_i(t) \delta u(t) dt, \\ s^-(t) \leq \delta u(t) \leq s^+(t). \quad (1)$$

Вводя на $[0, T]$ сетку с числом интервалов $N \gg m$ (обычно $m \sim 1 \div 10$, $N \approx 10^2 - 10^3$), мы получаем очень своеобразную задачу,

решение которой симплекс-методом может оказаться нерациональным. Напомним, что в симплекс-методе основным объектом является m -мерная грань. В задаче (1) подобная грань получается следующим образом: на $(N-m)$ счетных интервалах сетки Δ_i фиксировано в положении s^- или s^+ , и лишь на m интервалах Δ_i меняется в пределах $[s^-, s^+]$. Геометрические размеры подобной грани очень малы, и при $N \rightarrow \infty$ она стягивается в точку. Представляется естественным для задачи (1) строить методы решения, в которых основные геометрические объекты были бы связаны не с фиксированным числом m свободных (базисных) переменных, а с какими-то множествами на $[0, T]$, не зависящими, по существу, от шага сетки. Эти качественные соображения и были положены автором в основу итерационного метода, применявшегося при решении задач оптимального управления. Настоящий параграф содержит описание этого алгоритма, доказательство сходимости и основные расчетные формулы.

Итак, рассматривается следующая задача: заданы $(m+1)$ -мерные векторы $\{h_n\}_{n=1}^N$, X , e , и числа s_n^-, s_n^+ , $n=1, \dots, N$. Определено линейное отображение N -мерного прямоугольника $\sigma: \{s_n^- \leq s_n \leq s_n^+\}$ в выпуклый многогранник P в $(m+1)$ -мерном пространстве:

$$P: \quad x = X + \sum_{n=1}^N s_n h_n,$$

и нужно найти точку $\lambda e \in P$ с наименьшим λ и ее прообраз в σ . Предлагаемый алгоритм по основной идее близок к двойственному симплекс-методу. Ведущей идеей является эквивалентность сформулированной задачи задаче на минимакс: найти

$$\max_g \min_{x \in P} (x, g) \quad (\text{при условии } (g, e) = 1).$$

Напомним, что решение такой задачи само по себе определяет только значение λ , однако после того как λ и соответствующий вектор g найдены, определение прообраза λe в σ сводится к решению системы m линейных алгебраических уравнений (см. § 47).

Перейдем к описанию алгоритма.

1. Вычисления начинаются заданием вектора g , нормированного условием $(g, e) = 1$ (например, $g = e/\|e\|^2$), параметра η^* и вектора $\delta = \{\delta_0, \delta_1, \dots, \delta_m\}$. Число η^* и малый вектор δ определяют требуемую от решения точность. Итерационный метод дает нам не точное решение, а приближенное в следующем смысле: вместо соотношения

$$x = X + \sum_{n=1}^N s_n h_n = \lambda e$$

получим $|x - \lambda e| = |X + \sum s_n h_n - \lambda e| \leq \delta$ (это нужно понимать, как систему неравенств для всех компонент). Кроме того, величина λ

не будет точным минимумом, однако мы потребуем выполнения оценки типа

$$\eta = \frac{|\lambda - \lambda_{\min}|}{|\lambda_{\min}|} \leq \eta^*.$$

Кроме основных, имеющих четкий содержательный смысл, параметров η^* и δ , в алгоритм входят и другие. Для одних из них (ε , см. ниже) точная величина несущественна, и можно пользоваться грубыми практическими рекомендациями, для других это не так, и в алгоритм включаются правила подбора таких параметров. Все это будет в своем месте разъяснено.

2. Решается задача $\min(x, g)$. Для этого вычисляются:

$$a) \quad H_n^0 = (\overset{x \in P}{h_n}, g), \quad n = 1, 2, \dots, N,$$

$$b) \quad s_n = \begin{cases} s_n^- & \text{при } H_n^0 > 0, \\ s_n^+ & \text{при } H_n^0 \leq 0, \end{cases}$$

$$c) \quad x = X + \sum_{n=1}^N s_n h_n,$$

$$d) \quad F(g) = (x, g), \quad \lambda(x) = F(g).$$

Заметим, что $F(g) = \min_{x \in P} (x, g)$ дает оценку снизу для искомой величины: $\lambda_{\min} \geq F(g)$.

3. Выделяется некоторое подмножество M_ε «свободных» индексов n :

$$n \in M_\varepsilon \text{ экв. } |H_n^0| \leq \varepsilon \|h_n\|_G.$$

Поясним смысл нормы $\|\cdot\|_G$. В $(m+1)$ -мерном пространстве вводится «косоугольная» система координат, одной осью которой является прямая λe , а второй «осью» — m -мерная гиперплоскость G , ортогональная g . Всякий вектор x может быть представлен в виде

$$x = \lambda(x)e + x_G, \quad \text{причем } (x_G, g) = 0.$$

Легко находим это разложение:

$$\lambda(x) = (x, g), \quad x_G = x - \lambda(x)e$$

и, по определению, $\|x\|_G \equiv \|x_G\| = \|x - (x, g)e\|$ есть расстояние от x до оси λe , отсчитываемое в гиперплоскости, проходящей через x ортогонально g .

4. Решается задача

$$\min_{M_\varepsilon} \left\| X + \sum_{n=1}^N s_n h_n \right\|_G. \quad (2)$$

Эту запись нужно понимать в том смысле, что минимизация $\|X + \sum s_n h_n\|$ производится за счет изменения в интервалах $[s_n^-, s_n^+]$

только свободных переменных s_n , $n \in M_*$. Остальные s_n при этом фиксированы так, как они были получены в блоке 2, б). Задача минимизации (2) заслуживает отдельного описания. Здесь мы лишь отметим, что для этого используются алгоритмы покоординатного спуска и метода сопряженных градиентов. Задача (2) решается приближенно, и очень важным элементом алгоритма является правило, позволяющее судить о том, найден ли $\min \|x\|_g$ с нужной для дальнейшего точностью, или итерации следует продолжить. Важность этого правила связана с тем, что именно решение задачи (2) поглощает основную часть машинного времени. Выше фигурировал параметр ϵ . Он не имеет прямого содержательного смысла, поэтому ϵ не назначается, а вычисляется некоторым алгоритмом подбора. Блок 4 определяется еще и числом K итераций процесса минимизации (2). Это число задается, его величина не очень существенна (обычно $K \approx m$). После выполнения K итераций решения задачи (2), полученная точка

$$x = X + \sum_{n=1}^N s_n h_n$$

подвергается анализу.

5. Анализ должен дать ответ на следующие вопросы:

- а) не является ли величина ϵ слишком большой и не следует ли ее уменьшить?
- б) не получено ли уже решение с требуемой точностью?
- с) достаточно ли точно решена задача (2), и не следует ли продолжить итерации в блоке 4?

Для ответа на первый вопрос вычисляется

$$\eta = 2 \frac{|F(g) - (x, g)|}{|F(g) + (x, g)|}. \quad (3)$$

Если $\eta > \eta^*$, ϵ уменьшается (например, умножением на 0,75), и процесс вычислений направляется к блоку 2. При $\eta \leq \eta^*$ анализ продолжается. Если $|x - (x, g)e| = \|x_g\| \leq \delta$, задача считается решенной с необходимой точностью. В противном случае анализ продолжается. Наиболее сложно выяснение достаточности числа итераций при решении (2).

6. Вычисляются

$$y = x_g / \|x_g\|; \quad H_n^1 = (h_n, y)_g = (h_n - H_n^0 e, y),$$

$$n = 1, 2, \dots, N,$$

$$d = \frac{1}{\|x_g\|} \sum_{n \in M_*} H_n^1 \times \begin{cases} s_n - s_n^+ & \text{при } H_n^1 < 0, \\ s_n - s_n^- & \text{при } H_n^1 > 0. \end{cases}$$

Если $d < d^*$ (d^* — заданный параметр, $d^* < 1$, обычно в расчетах $d^* \approx 0,1 \div 0,3$), задача (2) считается решенной достаточно точно,

и переходим к блоку 7. В противном случае следует возвратиться в блок 4 и проделать очередной цикл итераций в задаче (2). Содержательный смысл этого критерия будет выяснен при доказательстве сходимости.

7. Пересчет вектора g . Образуется конструкция, содержащая неопределенный параметр

$$g(\alpha) = [1 - \alpha(e, y)]g + \alpha y, \quad (g(\alpha), e) \equiv 1, \quad (4)$$

и α определяется решением «одномерной» задачи

$$\max_{\alpha} \min_{x \in P} (x, g(\alpha)). \quad (5)$$

Она эквивалентна простейшей задаче линейного программирования: найти числа s_n , $s_n^- \leq s_n \leq s_n^+$, из условий

$$\min \left\{ \xi^0 + \sum_{n=1}^N s_n H_n^0 \right\}, \quad (6)$$

$$\xi^1 + \sum_{n=1}^N s_n H_n^1 = 0,$$

где $\xi^0 = (X, g)$, $\xi^1 = (X, y)_g$. Этот факт будет ниже доказан, а сейчас обратимся к решению задачи (6) *методом деления вилки*. Введем отображение σ в плоский многогранник π :

$$\pi: \quad z = \xi + \sum_{n=1}^N s_n H_n, \quad \xi = \{\xi^0, \xi^1\}, \quad H_n = \{H_n^0, H_n^1\}.$$

Для вектора $\beta = \{\beta_0, \beta_1\}$ определим $z(\beta)$ решением задачи $(z(\beta), \beta) = \min_{zx \in \pi} (z, \beta)$. Вектор $z(\beta)$ вычисляется точно так же, как и вектор x в блоке 2:

$$z(\beta) = \xi + \sum_{n=1}^N s_n H_n, \quad s_n = s_n^-(s_n^+) \quad \text{при} \quad (H, \beta) > 0 \quad (< 0).$$

Сначала задается $\beta = \{1; 0\}$, и $z(\beta)$ есть самая «нижняя» точка в π . При этом $z^1(\beta) > 0$. Этот вектор обозначим β^+ . Далее положим $\beta = \{0; 1\}$. Теперь $z(\beta)$ — самая левая точка в π . Если $z^1(\beta) > 0$, то π не пересекает ось ξ^0 , и задача (6) (а следовательно, и исходная задача) решения не имеет. Если $z^1(\beta) < 0$, то этот вектор β возьмем в качестве β^- . После этих двух нестандартных шагов процесс протекает уже однообразно: в качестве очередного вектора берется $\beta = \frac{1}{2}(\beta^+ + \beta^-)$, вычисляется $z(\beta)$, и текущий вектор β заменяет β^+ или β^- , в зависимости от знака $z^1(\beta)$. Процесс сходится со скоростью 2^{-k} , после ~ 20 итераций заканчивается, полагается $\alpha = \beta^0/\beta^1$, и вычисляется новый вектор g . Здесь же выясняется,

не является ли величина ϵ слишком малой. Для этого проверяется условие $\eta < \eta^*/2$, и, если оно выполнено, ϵ увеличивается (например, умножением на 1.25). Переходя к блоку 2, выполняем очередную итерацию процесса.

Обоснование метода. Прежде всего поясним смысл величины η . Если при решении задачи $\min \|X + \sum s_n h_n\|_G$ удастся получить точку $x = X + \sum s_n h_n$ такую, что $\|x\|_G = 0$, т. е. $x = \lambda(x)e \in P$, то величина $\lambda(x)$ даст нам точную оценку сверху для искомого λ_{\min} :

$$F(g) \leq \lambda_{\min} \leq \lambda(x). \quad (7)$$

Обычно в конце процесса достигаются малые, но все же отличные от нуля величины $\|x\|_G$, поэтому оценка λ_{\min} сверху не является точной. Тем не менее, ее можно считать практически достоверной. Это следует из простой оценки: обозначим через g^* вектор, определяющий гиперплоскость, опорную к P в искомой точке $\lambda_{\min}e$. Обычно в процессе итераций $g \rightarrow g^*$. Имеем

$$\begin{aligned} \lambda_{\min} = \min_{y \in P} (y, g^*) &\leq (x, g^*) = (x, g) + (x, g^* - g) = \\ &= (x, g) + (x_E, g^* - g) \leq \lambda(x) + \|x_E\| \|g^* - g\|. \end{aligned}$$

Используя очевидное соотношение $\|x_E\| = \|x\|_G \cos(g, e) = \|x\|_G(g, e)/\|g\| \|e\|$, получим окончательную оценку (уже точную):

$$F(g) \leq \lambda_{\min} \leq \lambda(x) + \frac{\|x\|_G \|g^* - g\|}{\|g\| \|e\|}. \quad (8)$$

Обычно в расчетах, кроме малости $\|x\|_G$, получаем и малые значения $\|g - g^*\|$ (т. е. $g \rightarrow g^*$). Поэтому оценка $\lambda_{\min} \leq (x, g)$ «практически достоверна». В то же время нетрудно построить пример задачи, в которой при $\|g\| \sim 1$ и сколь угодно малом значении $\|x\|_G$ в точном решении $\|g^*\|$ произвольно велика, и оценка имеет любую заданную погрешность. Это будут примеры «неустойчивых» задач, в которых малые изменения, например, X , приводят к большим изменениям λ_{\min} или даже к отсутствию решения. Заметим теперь, что если процесс осуществляется с $\epsilon = 0$, то свободными переменными будут лишь те, у которых $H_n^0 = (h_n, g) = 0$, поэтому в процессе решения задачи (2) точка x будет перемещаться в гиперплоскости G , ортогональной g , и величина $\lambda(x) = (x, g)$ не будет меняться. Если удастся получить точку с $\|x\|_G = 0$, то это будет точное решение, и $F(g) = \lambda_{\min} = \lambda(x)$. Однако реально вычисления производятся с $\epsilon > 0$ и $\lambda(x) > F(g)$. Величина η показывает, какой вырабатывается в процессе итераций разница $\lambda(x) - F(g)$; если она слишком велика, ϵ уменьшается. В то же время работать со слишком малым ϵ невыгодно: это замедляет скорость получения приближенного решения. Наиболее выгодным является максимальное ϵ , обеспечивающее все же заданную оценку $\eta \leq \eta^*$. Эти соображения

и определяют алгоритм подбора ε . Опыт вычислений (некоторые примеры будут ниже обсуждены) показывает, что обычно в конце процесса величина ε стабилизируется. Поэтому в дальнейшем мы для простоты будем считать ее постоянной. Введем некоторые полезные в дальнейшем геометрические объекты. Обозначим Q (точнее, $Q(g, \varepsilon)$) множество точек $x = X + \sum_{n \notin M} s_n h_n + \sum_{n \in M} s_n h_n$, причем в первой сумме s_n фиксированы в положениях $s_n^-(s_n^+)$ при $H_n^0 > 0 (< 0)$, а во второй s_n могут изменяться в пределах $[s_n^-, s_n^+]$. При $\varepsilon = 0$ множество Q было бы частью границы P , при $\varepsilon > 0$ Q может, вообще говоря, включать и некоторые близкие к границе внутренние точки P . В частности, задача (2) есть поиск точки $x \in Q$, ближайшей (в смысле метрики $\|\cdot\|_g$) к оси l_g . Мы будем рассматривать проекцию $(m+1)$ -мерного пространства в двумерное, определяемое векторами e и y . Именно, всякая точка $x \in P$ проектируется в $\{z^0, z^1\}$:

$$z^0 = (x, g); \quad z^1 = (x, y)_g.$$

Это проектирование можно осуществлять и другим способом. Если $x \in P$ имеет представление $x = X + \sum_n s_n h_n$, то ее проекция $z = \xi + \sum_n s_n H_n$ (ξ — проекция X). Эквивалентность этих двух способов следует из формул для $H_n = \{H_n^0, H_n^1\}$. Многогранник P проектируется в плоский многогранник π , а Q — в множество q :

$$z \in q: \quad z = \xi + \sum_{n \notin M} s_n H_n + \sum_{n \in M} s_n H_n.$$

Прежде всего докажем простую лемму, обосновывающую выбор параметра α при вычислении нового вектора g .

Лемма 1. *Задача $\max_{\alpha} \min_{x \in P} (x, g(\alpha))$ эквивалентна спроектированной задаче линейного программирования (6).*

Доказательство:

$$\begin{aligned} \max_{\alpha} \min_{x \in P} (x, g(\alpha)) &= \max_{\alpha} \min_{s^- \leq s \leq s^+} \left(X + \sum_{n=1}^N s_n h_n, g[1 - \alpha(e, y)] + \alpha y \right) = \\ &= \max_{\alpha} \min_{x \in P} \left\{ (X, g) + \sum_{n=1}^N s_n (h_n, g) + \alpha \left[(X, y) - (e, y)(X, g) + \right. \right. \\ &\quad \left. \left. + \sum_{n=1}^N s_n \{(h_n, y) - (e, y)(h_n, g)\} \right] \right\} = \\ &= \max_{\sigma} \min_s \left\{ \xi^0 + \sum_{n=1}^N s_n H_n^0 + \alpha \left[\xi^1 + \sum_{n=1}^N s_n H_n^1 \right] \right\} \end{aligned}$$

(были использованы соотношения $y_g = y$ и $h_g = h - H^0 e$). Рассмотрим отображение σ в π : $z = \xi + \sum_n s_n H_n$. В силу теорем

двойственности задача (6) эквивалентна задаче определения вектора $\tilde{g} = \{1; \alpha\}$ из условия $\max_{\alpha} \min_{z \in \pi} (z, \tilde{g})$. Этим и устанавливается эквивалентность задач (6) и (5) в том смысле, что они определяют одну и ту же величину α :

$$\max_{\alpha} F(g(\alpha)) = \max_{\alpha} \min_{x \in P} (x, g(\alpha)) = \max_{\alpha} \min_{z \in \pi} (z, \tilde{g}).$$

Полезно еще заметить, что при $g^* = \{1; 0\}$

$$\min_{z \in \pi} (z, g^*) = \min_s (\xi + \sum_n s_n H_n, g^*) = \min_{s^- \leq s \leq s^+} (\xi^0 + \sum_n s_n H_n^0) = F(g);$$

это следует из того, что

$$F(g) = \min_{x \in P} (x, g) = \min_s (X + \sum_n s_n h_n, g) = \min_s (\xi^0 + \sum_n s_n H_n^0).$$

Поскольку, как будет ниже показано, при $\|y\| \neq 0 \max_{\alpha} \min_{z \in \pi} (z, \tilde{g}) > \min_{z \in \pi} (z, g)$, то данная процедура обеспечивает эволюцию вектора g таким образом, что величина $F(g)$ монотонно растет. Для того чтобы получить отсюда сходимость к приближенному решению, нужно будет показать возможность лишь следующих двух случаев.

1. В процессе итераций $\|x_g\| \rightarrow 0$, т. е. получаем приближенное решение, так как даже при фиксированном ε можно получить оценку

$$\lambda(x) \leq F(g) + \sum_{n \in M} (s_n^+ - s_n^-) |H_n^0|.$$

Однако это очень грубая оценка, и мы просто предполагаем, что ε выбрано таким, что обеспечивается соотношение $\eta \leq \eta^*$.

2. Если $\|x\|_g \geq a > 0$, то каждая итерация процесса (пересчет g) сопровождается ростом $F(g)$ на величину, не меньшую некоторого $\Delta > 0$. Последнее противоречит оценке $F(g) \leq \lambda_{\min}$ (считаем, что решение задачи существует и λ_{\min} конечно). Справедливость этой альтернативы существенно связана со способом прерывания итераций (внутренних) при решении задачи минимизации (2).

Л е м м а 2. В задаче (6) для всех $n \notin M_{\varepsilon}$

$$|H_n^0| \geq \varepsilon |H_n^1|.$$

В самом деле,

$$\left| \frac{H_n^0}{H_n^1} \right| = \left| \frac{H_n^0}{(h_n, y)_g} \right| \geq \frac{\varepsilon \|h_n\|_g}{\|h_n\|_g \|y_g\|} = \varepsilon, \quad \text{так как } \|y_g\| = 1.$$

Л е м м а 3. Пусть в процессе итераций (2), т. е. при поиске $\min_{x \in Q} \|x\|_g$, получены точка $\tilde{x} \in Q$ и соответствующее ей \tilde{s} , удовле-

творяющие критерию $d \leq d^* < 1$. Тогда все точки $z \in q$ лежат правее прямой $z^1 = \mu$, где $\mu = (1 - d) \|\bar{x}\|_G$.

Доказательство. Вычислим $\mu = \min_{z \in q} z^1$. Очевидно,

$$\mu = \min_s \left\{ \xi^1 + \sum_{n \notin M} \bar{s}_n H_n^1 + \sum_{n \in M} s_n H_n^1 \right\} = \xi^1 + \sum_{n \notin M} \bar{s}_n H_n^1 + \min_s \sum_{n \in M} s_n H_n^1.$$

Точка $\bar{x} = X + \sum_{n \notin M} \bar{s}_n h_n + \sum_{n \in M} \bar{s}_n h_n$, причем $\bar{s}_n = s_n^-(s_n^+)$ при $n \notin M$.

Поэтому

$$\mu = \xi^1 + \sum_{n=1}^N \bar{s}_n H_n^1 + \sum_{n \in M} H_n^1 \begin{cases} \bar{s}_n - s_n^+ & \text{при } H_n^1 > 0, \\ \bar{s}_n - s_n^- & \text{при } H_n^1 < 0. \end{cases}$$

Таким образом, $\mu = (\bar{x}, y)_G - d \|\bar{x}\|_G = \|\bar{x}\|_G (1 - d)$, так как $y = \bar{x}_G / \|\bar{x}_G\|$ (см. также формулу для d).

Теперь рассмотрим строение границы π — ломаной γ . Вершинам γ соответствуют точки $s_n = \{s_n^+$ или $s_n^-\}$, гранями являются отрезки $(s_n^+ - s_n^-) H_n$, двум вершинам, принадлежащим одной и той же грани, соответствуют точки s , отличающиеся лишь одним s_n : для одной вершины $s_n = s_n^+$, для другой $s_n = s_n^-$.

Лемма 4. Самая низкая точка в π , являющаяся решением задачи $\min_{z \in \pi} (z^0)$, имеет координаты $\{F(g), \xi_1\}$, причем $\xi_1 > 0$. Она является проекцией точки $x \in Q$, вычисляемой в блоке 2 алгоритма.

Доказательство очевидно и опускается.

Введем теперь нумерацию граней γ , отсчитывая их влево от самой низкой точки π , как показано на рис. 79, с). Каждая грань порождена своим вектором H_n , и первой соответствует вектор H_{n_1} , второй — H_{n_2} , и т. д.

Лемма 5. Наклоны последовательных граней (т. е. $|H_{n_i}^0/H_{n_i}^1|$) образуют монотонно растущую с номером i последовательность (до тех пор, во всяком случае, пока эти грани расположены вправо от оси z_0 , при условии, что π пересекает ось z^0 , т. е. что на данном этапе расчета не обнаружено отсутствие решения задачи). Если i -я грань расположена в полосе $0 \leq z^1 \leq \mu$, то ее наклон не меньше ε .

Доказательство. Монотонность наклонов есть очевидное следствие выпуклости π . Рассмотрим первую грань. Для соответствующего индекса n_1 возможны два варианта:

1. $n_1 \notin M$. Тогда, по лемме 1, $|H_{n_1}^0/H_{n_1}^1| > \varepsilon$; то же справедливо, в силу монотонности наклонов, и для остальных граней в полосе $[0, \mu]$.

2. $n_1 \in M$. В этом случае первая грань целиком лежит в q ; следовательно, ее левая граница лежит правее прямой $z^1 = \mu$. Подобный же анализ теперь проводится для второй грани и т. д.

Таким образом, наклон первой же грани, левый конец которой падает в полуплоскость $z^1 < \mu$, обязательно превосходит ε . Лемма доказана.

Теорема 1. Пусть полный итерационный цикл приводит к переходу от вектора $g^{(v)}$ к $g^{(v+1)}$ (v — номер итерации), и пусть этот переход осуществляется в ситуации, когда приближенное решение задачи (2) $(\min_{x \in Q} \|x\|_g)$ дало точку $\tilde{x}^{(v)}$, причем $d \leq d^*$.

Тогда

$$F(g^{(v+1)}) > F(g^{(v)}) + \varepsilon \|\tilde{x}^{(v)}\|_g (1 - d^*). \quad (9)$$

Доказательство очевидным образом следует из леммы 4 и из того, что наклон границы γ на отрезке $0 \leq z^1 \leq \mu = \|\tilde{x}^{(v)}\|_g (1 - d)$ больше ε .

Формула (9) доказывает сформулированную выше альтернативу и, следовательно, сходимости к приближенному решению. Однако должна быть доказана еще и

Лемма 6. В итерационном процессе не может быть заикливания по признаку $d > d^*$.

Доказательство. Выясним геометрический смысл величины d , которую здесь будем обозначать точнее $d(x)$, $x \in Q$. Рассмотрим прямую, соединяющую (в гиперплоскости G , проходящей через x ортогонально g) начало координат с точкой x_g ; она имеет вид ξx_g . Найдем в Q точку y из условия: $\min_{y \in Q} \langle y, x_g \rangle$; проекция этой точки на прямую ξx_g есть $\xi_1 x_g$. Нетрудно теперь проверить, что $d(x) = (1 - \xi_1)$. Для дальнейшего важны следующие три почти очевидных факта:

- 1) $d(x)$ есть непрерывная функция x ;
- 2) если x^* есть точное решение задачи $\min_{x \in Q} \|x\|_g$, то $d(x^*) = 0$;

3) решение задачи (2) $(\min_{x \in Q} \|x\|_g)$ осуществляется сходящимся итерационным процессом; следовательно, $x \rightarrow x^*$, и в силу непрерывности $d(x) \rightarrow 0$. Лемма доказана.

Заметим, однако, что непрерывность $d(x)$ существенно связана с конечными размерами области Q . При бесконечных размерах Q , что бывает в задачах с односторонними ограничениями неизвестных $s_n \geq 0$, $d(x)$ уже не является непрерывной функцией. Этот случай нуждается в специальных дополнениях, которые подробно изложены в [94]. Здесь мы их не приводим. Заметим, что существует очень простой способ обойти это затруднение, заменив односторонние ограничения $0 \leq s_n$ на двусторонние $0 \leq s_n \leq S$, где S — достаточно большое число, так что задача от этого не меняется. Выбор такого большого числа был бы нетруден, однако простота решения — обманчива: при большом S и, следовательно, при больших размерах Q достижение ситуации $d(x) \leq d^*$ потребовало бы

очень точного решения задачи $\min_{x \in \pi} \|x\|_g$, что связано с большими затратами машинного времени. Обратимся теперь к рис. 79, иллюстрирующему некоторые положения. На рис. 79, а) изображен многогранник π , не пересекающийся с осью z^0 . В этом случае

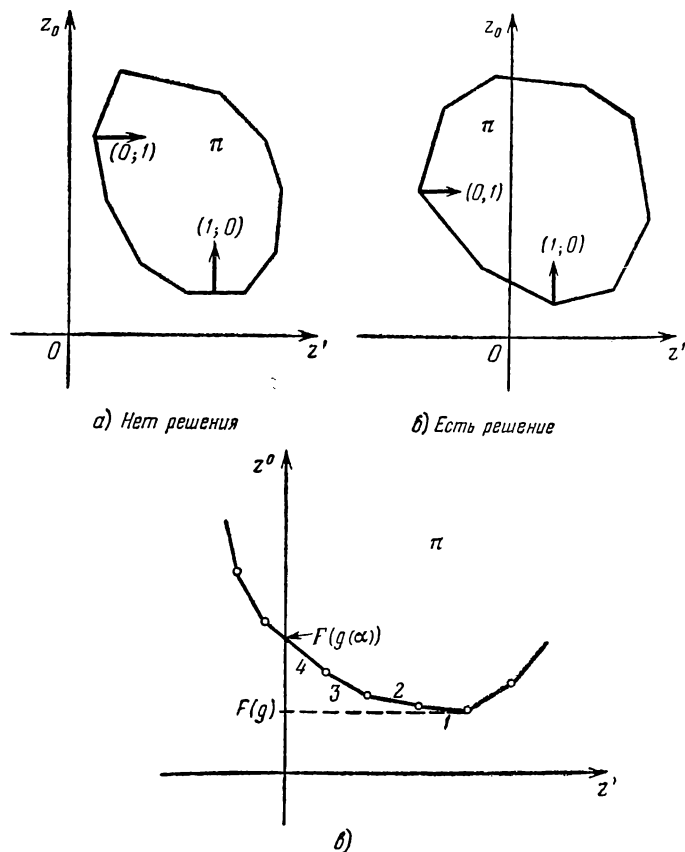


Рис. 79.

исходный многогранник P не пересекается с прямой l_g , и задача не имеет решения. Этот факт обнаруживается при решении задачи (6), если самая левая в π точка, решающая задачу $\min_{z \in \pi} (z, \tilde{g})$ при $\tilde{g} = \{0; 1\}$, расположена в правой полуплоскости. Рис. 79, б) иллюстрирует процесс «деления вилки» при решении спроектированной задачи. Показаны последовательно получаемые векторы \tilde{g} и соответствующие им точки в π , реализующие $\min_{z \in \pi} (z, \tilde{g})$.

Рис. 80, а) поясняет геометрический смысл величины $d(x)$. Наконец, рис. 80, б) поясняет, почему при больших размерах Q ситуация $d(x) \leq d^*$ требует очень точного приближения точки x к точке минимума x^* величины $\|x\|_G$, $x \in Q$.

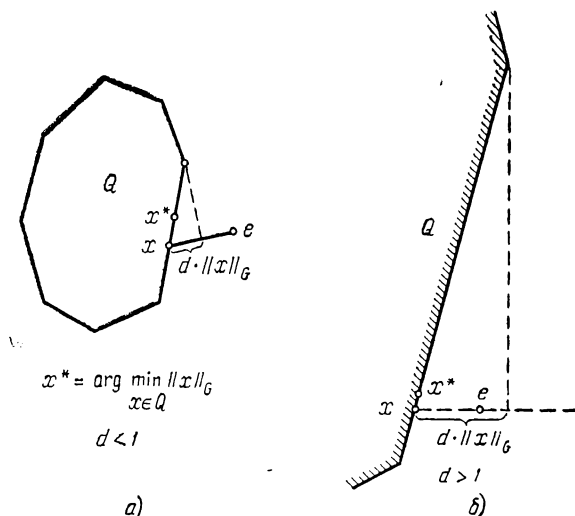


Рис. 80.

Алгоритм решения задачи $\min_{x \in Q} \|x\|_G$. Прежде всего запишем задачу в терминах s_n :

$$\min_s \left\| X' + \sum_{n \in M} s_n h_n \right\|_G, \quad \text{где} \quad X' = X + \sum_{n \notin M} s_n h_n. \quad (10)$$

Используются два способа решения (10): метод покоординатного спуска и метод сопряженных градиентов. Большое число переменных s_n , $n \in M$, и наличие ограничений $s_n^- \leq s_n \leq s_n^+$, а также специфика задачи, состоящая в том, что векторы h_n для близких индексов n близки между собой, потребовали введения некоторых дополнений в эти стандартные алгоритмы. Прежде всего вводится последовательность признаков π_n , $n=1, \dots, N$, причем $\pi_n=0$ для $n \notin M$ и $\pi_n > 0$ для $n \in M$. Решение задачи (10) начинается методом покоординатного спуска. Управление процессом осуществляется с помощью двух признаков ξ_1 , ξ_2 , смысл которых будет разъяснен ниже.

1. Если $\pi_n \neq 0$, то $\pi_n := 2$ ($n=1, \dots, N$); кроме того, $\xi_1 := 0$.

2. $\xi_2=0$. Для $n=1, \dots, N$ и $\pi_n=2$ проводится изменение переменной s_n с целью минимизации $\|x\|_G$. Для этого вычисляется δ из условия

$$\min_s \|x + \delta h_n\|, \quad \text{т. е.} \quad \delta = -(x, h_n)_G / \|h_n\|_G^2.$$

Далее учитываются ограничения на s_n :

$$\delta s_n = \begin{cases} \min(\delta, s_n^+ - s_n) & \text{при } \delta > 0, \\ \max(\delta, s_n^- - s_n) & \text{при } \delta < 0. \end{cases}$$

После этого пересчитываем $x := x + \delta s_n h_n$, $s_n := s_n + \delta s_n$. Кроме того, если s_n вышло на границу (s_n^- или s_n^+), то переменное s_n временно исключается из процесса тем, что $\pi_n := 1$, а сам факт выхода на границу отмечается изменением признака $\xi_2 := 1$.

3. После окончания цикла 2 мы можем встретиться со следующими комбинациями признаков:

- $\xi_1=0$; $\xi_2=1$: тогда полагаем $\xi_1=1$ и переходим к 2.
- $\xi_1=1$; $\xi_2=1$: в этом случае тоже переходим к 2.
- $\xi_1=1$; $\xi_2=0$: переходим к 1.
- $\xi_1=0$; $\xi_2=0$: в этом случае переходим к решению задачи (10) методом сопряженных градиентов.

Поясним смысл такого управления. Временное исключение переменных имеет следующую цель: пусть переменная s_n вышла на s_n^+ ; можно предположить, что и в дальнейшем в целях минимизации $\|x\|_G$ эту переменную следует увеличить, но это уже неосуществимо. Поэтому лучше не тратить времени на вычисление для нее величины δ . Но этот вывод нестрог, так как после выхода s_n на s_n^+ точка x за счет изменения других s_n изменилась и, может быть, нужно менять s_n в другую сторону. Поэтому различаются два случая (с и d) при $\xi_2=0$: если в цикле 2 не было случаев выхода на границу, но часть переменных была исключена ($\xi_1=1$), то следует снова включить все s_n в работу. Если же $\xi_2=0$ при $\xi_1=0$, то это означает, что в решении задачи (10) ограничения s^- , s^+ перестали играть роль (тоже, может быть, лишь временно) и задача (10) стала задачей минимизации без ограничений. В этой ситуации метод покоординатного спуска становится неэффективным, и целесообразно перейти к методу сопряженных градиентов. Правда, перед этим переходом полученная ситуация подвергается анализу (см. выше). Если анализ показывает необходимость более точного решения задачи (10), это делается следующим образом.

4. $\pi_n := 2$, если $\pi_n \neq 0$ и $s_n^- < s_n < s_n^+$. Таким образом, будут варьироваться только переменные s_n , не вышедшие на границу.

5. Для $n=1, \dots, N$ и $\pi_n=2$ полагаем $c_n = -H_n^1 (H_n^1 = (x, h_n)_G = \partial \|x\|_G^2 / \partial s_n)$ вычислены в блоке анализа).

$$6. v = \sum_{\pi_n=2} c_n h_n.$$

Это есть направление спуска для $\|x\|_G^2$. Находим предварительный шаг спуска Δ решением задачи $\min \|x + \Delta v\|_G^2$, т. е. $\Delta = -\frac{(x, v)_G}{(v, v)_G}$.

Далее находится действительный шаг спуска, учитывающий ограничения переменных:

$$\delta = \frac{1}{c_n} \begin{cases} \min_{\pi_n=2} \min \{\Delta c_n, s_n^+ - s_n\} & \text{при } \Delta c_n > 0, \\ \max_{\pi_n=2} \max \{\Delta c_n, s_n^- - s_n\} & \text{при } \Delta c_n < 0. \end{cases}$$

7. Пересчитываются вектор x и переменные s_n :

$$x := x + \delta v; \quad s_n := s_n + \delta c_n \quad (\pi_n = 2).$$

Здесь также отмечается факт выхода на границу, и если он имел место, то соответствующая переменная s_n исключается ($\pi_n = 1$), вычисляются новые H_n^1 (для $\pi_n = 2$), и переходим к пункту 5.

8. Если выхода на границу не было, пересчитывается вектор $\{c_n\}$ процедурой ортогонализации (см. § 51) (все вычисления ведутся лишь при $\pi_n = 2$):

а) вычисляются H_n^1 ,

б) $z = -\sum H_n^1 h_n$,

в) $c_n := -H_n^1 + \frac{(z, v)_G}{(v, v)_G} c_n$.

Далее переходим к пункту 6.

Выше описан основной цикл метода сопряженных градиентов. Таких циклов делается $\sim m$, после чего происходит снова возврат к покоординатному спуску (п. 1). Поясним, почему сразу не используется метод сопряженных градиентов. В этом методе все переменные s_n изменяются одновременно, и шаг определяется наименьшим расстоянием одной из переменных до своей границы s^- (s^+). Пусть этот шаг определяется переменной s_j . Однако в процессе участвуют векторы h_n , близкие к h_j (напомним, что h_n суть сеточное представление непрерывной функции $w(t)$ в (1)). Поэтому многие переменные s_n лишь немного не дотянут до своих границ. На следующем цикле процесса на границу выйдет одна из этих переменных, причем смещение δ будет очень малым, затем еще одна и т. д. В целом процесс будет неэффективен, так как каждая итерация метода сопряженных градиентов требует значительных предварительных вычислений.

Опыты вычислений. Итерационный метод решения задач линейного программирования систематически использовался автором начиная с 1963 г. Разумеется, опыт эксплуатации метода приводил к различным усовершенствованиям, и выше алгоритм изложен таким, каким он сложился ко времени написания книги. (Впрочем, в изложении опущены некоторые несущественные де-

тали). В основном метод использовался при решении задач оптимального управления, однако были проведены эксперименты по проверке его и в других условиях, близких к тем, которые характерны для задач экономического содержания. В частности, в [90] подробно описан опыт решения задачи очень высокой размерности ($N \approx 11\,000$, $m \approx 5500$) со слабо заполненной матрицей ($\approx 30\,000$ ненулевых элементов). В таблицах 1—3 показан характерный пример решения задачи линейного программирования ($N=130$, $m=90$, h_n^* заполнялись случайными числами, равномерно распределенными на $[0, 1]$ для $i=0$, и на $[-5, 5]$ для $i=1, \dots, m$).

Т а б л и ц а 1

$$\eta^*=0.002$$

ν	$F(g)$	$\lambda(x)$	$\ x\ _G$	n	i	$s \cdot 10^4$
1	0	0	117	3	2	10
2	0,543	0,547	111	4	2	8,2
3	1,212	1,230	107	6	4	6,7
4	1,954	1,972	106	7	4	5,5
21	7,215	7,228	89	15	4	1,0
22	7,258	7,271	80	20	6	1,2
23	7,731	7,750	79	21	6	1,4
24	7,782	7,800	76	21	7	1,1
41	12,980	13,005	56	35	6	1,1
42	13,161	13,189	55	36	12	1,3
43	14,003	14,031	52	36	15	1,5
44	14,084	14,115	45	40	15	1,7
61	17,260	17,297	23	50	20	1,7
62	17,359	17,396	23	50	15	1,4
63	17,392	17,428	24	49	15	1,2
64	17,414	17,448	23	52	20	1,3
65	17,496	17,531	20	52	25	1,5
81	19,554	19,598	10,6	65	31	1,9
82	19,596	19,640	10,5	66	41	1,5
83	19,621	19,660	22	56	45	1,3
84	19,627	19,666	10,3	67	59	1,4
101	20,946	20,991	5,4	77	72	5,7
102	21,065	21,110	5,4	76	95	4,6
103	21,117	21,162	5,3	78	55	3,8
104	21,153	21,200	4,9	79	91	3,1
111	21,322	21,363	3,0	80	110	1,5
112	21,373	21,414	2,6	79	106	1,8
113	21,389	21,429	2,6	80	80	2,0
114	21,406	21,446	2,4	81	105	2,3
115	21,416	21,456	1,8	82	141	2,7
116	21,422	21,463	1,6	85	136	3,1
122	21,576	21,617	0,3	86	186	7,1
123	21,582	21,623	0,28	87	246	8,2
124	21,613	21,655	0,26	87	271	9,4
125	21,619	21,661	0,04	88	120	10,8

Т а б л и ц а 2

$$\eta^* = 0,02$$

ν	$F(g)$	$\lambda(x)$	$\ x\ _G$	n	i	$\varepsilon \cdot 10^4$
1	0	0,003	117	3	2	10
2	0,543	0,566	109	5	2	12
3	1,456	1,526	104	9	4	13
4	2,649	2,697	104	9	2	11
15	12,987	13,251	55	36	12	18
16	13,797	14,085	45	41	18	21
17	14,302	14,596	42	44	14	24
18	15,323	15,691	31	46	24	28
31	19,612	20,001	9,2	71	67	16
32	19,685	20,084	8,1	71	52	18
33	19,963	20,361	7,7	72	53	21
34	20,122	20,556	6,7	75	65	24
35	20,310	20,729	6,4	77	70	20
42	21,038	21,474	1,6	83	133	27
43	21,076	21,504	0,78	86	209	31
44	21,105	21,535	0,63	87	226	36
45	21,171	21,612	0,45	87	186	41
46	21,229	21,660	0,04	88	120	54

Т а б л и ц а 3

$$\eta^* = 0,01$$

ν	$F(g)$	$\lambda(x)$	$\ x\ _G$	n	i	$\varepsilon \cdot 10^4$
21	11,126	11,248	58	32	9	11
22	11,980	12,13	56	34	9	13
23	12,94	13,09	55	35	10	14
24	13,80	13,93	53	36	12	17
35	16,79	16,98	24	55	18	10
36	17,25	17,45	20	54	21	8,3
38	18,40	18,63	15	58	35	9,6
39	18,70	18,92	15	59	46	11
40	18,81	19,03	14	61	35	13
53	20,96	21,22	4,7	80	126	12
54	21,09	21,35	3,8	81	143	14
55	21,13	21,39	2,8	82	124	16
56	21,19	21,44	1,9	81	123	18
57	21,22	21,49	1,0	86	197	21
58	21,24	21,51	0,63	86	195	24
59	21,35	21,62	0,25	87	220	23
60	21,383	21,663	0,002	88	300	20

В таблицах показана эволюция в процессе решения следующих величин: ν — номер итерации, $F(g)$, $\lambda(x)$, $\|x\|_G$, n — число индексов в множестве M_ϵ , i — число итераций метода сопряженных градиентов при решении задачи (2), (в этих расчетах метод по координатного спуска не использовался), и, наконец, ϵ . В этой задаче $e = \{1, 0, \dots, 0\}$. Расчеты отличались лишь различными требованиями к точности определения: $\eta^* = 0,002$ (табл. 1), $\eta^* = 0,02$ (табл. 2) и $\eta^* = 0,01$ (табл. 3). Отметим следующие характерные черты работы алгоритма.

1. Объем вычислений существенно связан с величиной η^* .

2. Во всех трех расчетах была получена практически одна и та же величина $\lambda(x) = 22,660 - 22,663$, в то же время оценки снизу отличаются заметно больше. Это — типичная ситуация, отмечавшаяся во всех других экспериментах: при различных требованиях к точности (η^*), алгоритм (при существенно разных затратах машинного времени) дает более или менее одно и то же решение, отличается лишь оценка снизу, т. е. гарантированная точность решения. Поэтому в задачах типа (1), учитывая, что сама по себе эта задача приближенная и особая точность в ее решении не нужна, обычно $\eta^* \approx 0,1$.

3. Значительный перерасход машинного времени был связан с тем, что метод сопряженных градиентов был не конечным, как это утверждает теория, а лишь итерационным. Это связано с влиянием ошибок округления, не учитываемых теорией.

Особенно сильно это сказывается при большом (≈ 90) числе переменных s_n , участвующих в задаче (2). Подчеркнем еще раз, что этот пример расчета не характерен для аппроксимаций континуальных задач линейного программирования (1). В последних обычно $m \approx 1 \div 10$ и число итераций $\approx m$. Проводились эксперименты с очень малыми значениями ϵ , при которых получается практически точное решение. Хотя в таких расчетах в задаче (2) участвует всегда небольшое ($\approx m$) число переменных и объем вычислений на каждую такую задачу невелик, в целом процесс решения резко замедлялся, число итераций оказывалось слишком большим. Эти опыты подтверждают, что задачи, происходящие из (1), имеют свою специфику и ее следует использовать.

§ 49. Итерационный метод решения специальной задачи квадратического программирования

Реализация метода проекции градиента в § 18 привела к необходимости решения следующей вспомогательной задачи: найти числа s_n , $n = 1, 2, \dots, N$ из условий

$$\min \sum_{n=1}^N \left(s_n h_n^0 + \frac{1}{2S} s_n^2 \right), \quad (1)$$

$$X^i + \sum_{n=1}^N s_n h_n^i = 0, \quad i = 1, 2, \dots, m, \quad (2)$$

$$s_n^- \leq s_n \leq s_n^+, \quad n = 1, \dots, N. \quad (3)$$

Использовался алгоритм приближенного решения задачи (1)–(3), очень близкий по основным идеям к итерационному алгоритму § 48 и переходящий в него при $S \rightarrow \infty$, когда задача (1)–(3) превращается в задачу линейного программирования. Поэтому здесь будут приведены лишь основные формулы алгоритма, а некоторые детали, по существу тождественные соответствующим деталям алгоритма § 48, будут опущены.

Входной информацией, определяющей работу алгоритма, являются X^i , s_n^- , s_n^+ , S , h_n^i , начальное значение $(m+1)$ -мерного вектора $g = \{1, g_1, \dots, g_m\}$, число η^* , характеризующее точность решения по значению минимизируемой формы (1), числа Δ_i , определяющие заданную точность выполнения условий (2).

Удобно будет пользоваться некоторыми простыми геометрическими объектами: прямоугольник σ в N -мерном пространстве $(\{s_n^- \leq s_n \leq s_n^+\}, n=1, \dots, N)$, $(m+1)$ -мерный вектор $e = \{1, 0, \dots, 0\}$, $X = \{0, X^1, \dots, X^m\}$. Образ σ при отображении в $(m+1)$ -мерное пространство, обозначаемый P :

$$x^0 = \sum_{n=1}^N \left(s_n h_n^0 + \frac{1}{2S} s_n^2 \right), \quad x^i = X^i + \sum_{n=1}^N s_n h_n^i, \quad (4)$$

$$i = 1, 2, \dots, m.$$

Задачу (1)–(3) будем интерпретировать как задачу поиска точки $\lambda e \in P$ с минимальным $\lambda = \Lambda$ и ее прообраза в σ .

I. Решается задача

$$\min_{x \in P} (x, g) = \min_{s^- \leq s \leq s^+} \left\{ \sum_{n=1}^N \left[s_n (h_n, g) + \frac{1}{2S} s_n^2 \right] + (X, g) \right\}.$$

При этом вычисляются

$$s_n = \begin{cases} s_n^-, & \text{если } -S(h_n, g) < s_n^-, \\ s_n^+, & \text{если } -S(h_n, g) > s_n^+, \\ -S(h_n, g) & \text{— в остальных случаях,} \end{cases}$$

и вектор $\check{x} = \{x^0, x^1, \dots, x^m\}$ по формулам (4).

Вычисляется величина $F(g) = (x, g)$, являющаяся для Λ оценкой снизу: $F(g) \leq \Lambda = \lambda_{\min}$.

II. По заданному числу ε (алгоритм подбора которого в точности совпадает с алгоритмом подбора ε в § 48) вычисляются новые границы допустимых изменений переменных s_n : $s_n^- \leq c_n^- \leq s_n \leq c_n^+ \leq s_n^+$. В дальнейшем числам s_n будет разрешено меняться в пределах $[c_n^-, c_n^+]$, и назначение этих отрезков делается так, чтобы соответствующее перемещение точки $x(s)$ (см. (4)) происходило почти в гиперплоскости $(x - \check{x}, g) = 0$. Заметим, что

$$\frac{\partial (x(s), g)}{\partial s_n} = \frac{1}{S} s_n + (h_n, g).$$

Определим теперь область изменения s_n , в которой

$$\left| \frac{\partial (x(s), g)}{\partial s_n} \right| \leq \varepsilon \|h_n\|_E, \text{ где } \|h\|_E = \left\{ \sum_{i=1}^m (h_n^i)^2 \right\}^{1/2}.$$

Эта область (« ε -отрезок») имеет вид

$$S[-(h_n, g) - \varepsilon \|h_n\|_E] \leq s_n \leq S[-(h_n, g) + \varepsilon \|h_n\|_E].$$

ε -отрезок может лежать целиком левее s_n^- , тогда $c_n^- = c_n^+ = s_n^-$ может лежать правее s_n^+ , и тогда $c_n^- = c_n^+ = s_n^+$. В этих случаях переменная s_n оказывается «закрепленной». Если же « ε -отрезок» пересекается с $[s_n^-, s_n^+]$, то $[c_n^-, c_n^+]$ есть их пересечение.

III. Решается задача

$$\min_{c^- \leq s \leq c^+} \|x(s)\|_E. \quad (5)$$

Другими словами, решается задача минимизации квадратичной формы:

$$\min_{c^- \leq s \leq c^+} \sum_{i=1}^m \left\{ X^i + \sum_{n=1}^N s_n h_n^i \right\}^2.$$

Здесь используется та же самая комбинация покоординатного спуска и метода сопряженных градиентов, что и в § 48 (закрепленные переменные, разумеется, в процессе минимизации не участвуют). После некоторого количества итераций полученная точка $x(s)$ подвергается анализу.

IV. Анализ. 1. Вычисляется величина

$$\eta = 2 \frac{|(x, g) - F(g)|}{|(x, g) + F(g)|}.$$

При $\eta > \eta^*$ уменьшается ε , и вычисления продолжают переходом к 1. В противном случае анализ продолжается.

2. Проверяются условия $|x^i| \leq \Delta_i$. Если все они выполнены, задача решена с требуемой точностью.

3. Выясняется, решена ли задача (5) с необходимой для дальнейшего точностью или итерации III следует продолжить. Для этого находятся

$$H_n^1 = (h_n, x)_E = \sum_{i=1}^m (h_n^i, x^i),$$

и величина

$$d = \frac{1}{\|x\|_E^2} \sum_{n=1}^N H_n^1 \times \begin{cases} s_n - c_n^- & \text{при } H_n^1 > 0, \\ s_n - c_n^+ & \text{при } H_n^1 < 0. \end{cases}$$

Если $d > d^*$ ($d^* < 1$ — задано), вычисления возвращаются в блок III для уточнения решения (5) (см. § 48). При $d \leq d^*$ переходим к пересчету g .

V. Новый вектор g ищется в форме

$$g(\alpha) = g + \alpha x_E \quad (x_E = \{0, x^1, \dots, x^m\}).$$

Параметр α определяется задачей

$$\max_{\alpha} F[g(\alpha)] = \max_{\alpha} \min_{z \in P} (g(\alpha), z). \quad (6)$$

Переходя к s -представлению, получаем для α задачу

$$\max_{\alpha} \min_{s^- \leq s \leq s^+} \left\{ \frac{1}{2S} \sum_n s_n^2 + \sum_n s_n (h_n, g) + \right. \\ \left. + \alpha \sum_n (x_E, h_n) + (X, g) + \alpha (X, x_E) \right\}. \quad (6^*)$$

Теорема. Задача (7) эквивалентна простейшей задаче квадратического программирования

$$\min_{s^- \leq s \leq s^+} \left\{ Z^0 + \sum_{n=1}^N \left(s_n H_n^0 + \frac{1}{2S} s_n^2 \right) \right\}$$

при условии

$$Z^1 + \sum_{n=1}^N s_n H_n^1 = 0,$$

где $H_n^0 = (h_n, g)$, $Z^0 = (X, g)$, $Z^1 = (X, x_E)$.

Доказательство очевидно и опускается. Сама же задача решается алгоритмом деления вилки, совпадающим, по существу, с соответствующим алгоритмом в § 48. Заметим, что здесь же может выясниться отсутствие решения задачи (6*) и, следовательно, исходной задачи (1)–(3). После вычисления α и нового вектора g переходим к 1. Обоснование сходимости алгоритма может быть проведено почти дословно теми же рассуждениями, которые

использовались в § 48 для доказательства сходимости алгоритма линейного программирования. При $\epsilon=0$ алгоритм превращается в классический, сходящийся со скоростью геометрической прогрессии, алгоритм строго выпуклого программирования. При этом исключается задача (5), решение которой поглощает большую часть машинного времени. Естественно возникает вопрос: а нельзя ли обойтись этим классическим вариантом алгоритма? Был проведен вычислительный эксперимент. Задача о спуске космического аппарата (§ 37, расчет 4) решалась методом проекции градиента, использующим алгоритм решения задачи квадратичного программирования. Была сделана попытка повторить этот расчет с единственным изменением: $\epsilon=0$. Затратив на 50% больше машинного времени, чем занял весь расчет № 4, удалось выполнить лишь шесть итераций, причем в наиболее легких ситуациях, когда условия задачи грубо нарушены и решения задачи квадратичного программирования не существует. Кроме того, эти итерации проводились в условиях малой размерности задачи $m=2$, тогда как большая часть расчета 4 проводилась при $m=9$. Этот пример лишний раз показывает, какие проблемы могут возникнуть на низшем уровне технологического оформления задачи, и почему автор так скептически относится к работам, намечающим «принципиальные пути» решения задачи и не доводящим дело до ее фактического решения. В этом же расчете 4, после нескольких итераций при $\epsilon > 0$ в ситуации $m=5$ была сделана попытка продолжить расчет с $\epsilon=0$, и выданы подробные данные, иллюстрирующие ход итерационного процесса решения задачи

$$\max_g \min_{x \in P} (x, g).$$

В табл. 1 показаны в зависимости от номера итерации γ величины $\min (x, g)$ и $\|x\|_E$. Все происходит так, как предписывает теория: $\min (x, g)$ монотонно растет, $\|x\|_E$ стремится к нулю (немонотонно, естественно). Только уж очень медленно! Разумеется, такие эксперименты следует трактовать осторожно. Не исключено, что можно ускорить сходимость классического алгоритма строго выпуклого программирования, используя, например, метод сопряженных градиентов при решении задачи $\max_g F(g)$, где $F(g) \equiv \min (x, g)$. Этот рецепт звучит убедительно и, кажется, решает проблему, так как $F(g)$ вычисляется очень просто и точно, а размерность g не так уж велика (в расчетах, о которых шла речь, не более 10). Однако эффективность метода сопряженных градиентов существенно опирается на гладкость вторых производных $F(g)$. В задаче (1)–(3) можно ручаться за непрерывность $F(g)$, ее дифференцируемость, и, видимо, непрерывность первых производных. Дальнейшая гладкость $F(g)$ — сомнительна. Поэтому нужно экспериментировать.

Т а б л и ц а 1

ν	$\min(x, g)$	$\ x\ _E$	ν	$\min(x, g)$	$\ x\ _E$
0	0,8912183	0,81757	100	0,9740760	0,062873
1	0,9174131	0,51412	101	0,9740770	0,045321
2	0,9440996	0,76911	102	0,9740780	0,062715
3	0,9460493	0,35608	103	0,9740800	0,062558
4	0,9495225	0,66098	104	0,9740800	0,062558
5	0,9508411	0,31650	105	0,9740810	0,045095
6	0,9529874	0,57525	130	0,9741052	0,060628
40	0,9701974	0,08634	131	0,9741061	0,043761
41	0,9702791	0,071939	132	0,9741070	0,060477
42	0,9703871	0,085764	133	0,9741080	0,043653
43	0,9704701	0,071559	134	0,9741089	0,060329
44	0,9705774	0,085198	135	0,9741098	0,043546
45	0,9706619	0,071175	178	0,9741476	0,057448
70	0,9729668	0,079902	179	0,9741484	0,041491
71	0,9730468	0,066844	180	0,9741493	0,057332
72	0,9731464	0,079536	181	0,9741501	0,041407
73	0,9732262	0,066699	182	0,9741509	0,057218
74	0,9733300	0,079150	183	0,9741518	0,041325
75	0,9734100	0,066569	184	0,9741526	0,057105

Предостережение. Опыт решения задач оптимального управления методом проекции градиента еще не очень велик, и некоторые детали не совсем ясны. Автор хотел бы предупредить читателя о возможных осложнениях. Прежде всего, не очень ясен вопрос о назначении величины η^* , задающей требуемую относительную точность решения по значению минимизируемой формы (1). В задаче линейного программирования (при $S = \infty$) форма (1) имеет простой содержательный смысл — это значение $\delta F_0 [\delta u(\cdot)]$, и назначение $\eta^* = 0,1$ (0,2 или 0,05, если угодно) в особых разъяснениях не нуждается. В задаче квадратичного программирования форма (1) уже не имеет такого простого значения, часто ее значение бывает много больше $\delta F_0 = \sum s_n h_n^0$, поэтому вопрос о назначении η^* осложняется. Не исключено, что следует разработать алгоритм подбора η^* , реагирующий на результаты вычислений. Во всяком случае, опыт показал, что следует назначать существенно меньшие значения η^* , чем в задаче линейного программирования. Хотя практически удавалось (и без особого, в сущности, труда) найти η^* , обеспечивающее успешное течение процесса в целом, автор затрудняется сформулировать естественные принципы его выбора, на основе которых можно было бы построить и алгоритм вычисления η^* , подобно тому как был построен алгоритм вычисления ϵ по η^* . Видимо, поэтому метод проекции градиента проигрывал методу последовательной линеаризации на последней стадии расчета; так, расчет 2 (§ 37), продолженный дальше, не привел к улуч-

шению результата: значение минимизируемого функционала стабилизировалось на величине 521,5—522,0, тогда как расчет при $S=\infty$ привел к значению 520,3. Упомянутый выше экспериментальный подбор η^* осуществлялся очень просто, но существенно опирался на имеющийся расчет 1 с $S=\infty$: назначив некоторое η^* и не получив результатов, сопоставимых с результатами расчета 1, автор предположил в качестве возможной причины неудачи чрезмерную грубость решения задачи (1)—(3) и уменьшил η^* . В результате нескольких проб было найдено η^* , при котором процесс решения задачи (расчет 2) протекал в основном аналогично расчету 1. В конце расчета, на стадии уточнения решения, видимо, следовало скорректировать значение η^* . Этого, однако, сделано не было.

Вероятно, можно задавать «естественные» значения $\eta^* \simeq 0,1$, если использовать другую формулу для η :

$$\eta = \frac{|F(g) - (x, g)|}{\left| \sum_n s_n h_n^0 \right|} = \frac{|F(g) - (x, g)|}{|\delta F_0|}.$$

Второй вопрос, возникающий при реализации метода, связан с выбором единиц измерения для разных компонент управления. Мы не будем рассматривать общей ситуации, но попробуем разобратся в проблеме на примере той же задачи о спуске космического аппарата. Управлением являются функция $u(\cdot)$ и параметр T . Пусть единицы измерения $u(\cdot)$ — фиксированы, и вопрос стоит только о выборе единиц измерения T . Запишем задачу (1)—(3) в произвольных единицах измерения, выделив для наглядности переменную s (без индекса), имеющую смысл δT :

$$\min \left\{ \sum_n \left(s_n h_n^0 + \frac{1}{2S} s_n^2 \right) + \left(s h^0 + \frac{1}{2S} s^2 \right) \right\} \quad (7)$$

при условии

$$\sum_n s_n h_n^1 + s h^1 = 0$$

(ради упрощения ограничимся случаем $m=1$, ограничения типа $s^- \leq s \leq s^+$ тоже не будем выписывать). Произведем в задаче замену переменных $T' = \mu T$ (выберем другую единицу измерения T). Тогда задача примет вид

$$\min \left\{ \sum_n \left(s_n h_n^0 + \frac{1}{2S} s_n^2 \right) + \left[s' \frac{1}{\mu} h^0 + \frac{1}{2S} (s')^2 \right] \right\} \quad (7^*)$$

при

$$\sum_n s_n h_n^1 + s' \frac{1}{\mu} h^1 = 0.$$

Разумеется, границы изменения переменной s' должны быть пересчитаны в соответствии с соотношением $s' = \delta T' = \mu \delta T = \mu s$: $(s')^{\pm} = \mu s^{\pm}$. Заметим, что при $S = \infty$ мы имели дело с задачей линейного программирования, решение которой не зависит от μ в том смысле, что, решив задачу (7) и задачу (7*), получим значения s и s' , связанные соотношением $s' = \mu s$. Однако при $S < \infty$ такой инвариантности уже нет, решения задач (7) и (7*) могут существенно отличаться друг от друга. В самом деле, решим (7*), отбрасывая для простоты условия $s^- \leq s \leq s^+$. Вводя множитель Лагранжа λ , получим

$$s_n = -S(h_n^0 + \lambda h_n^1); \quad s' = -S\left(\frac{1}{\mu} h^0 + \frac{\lambda}{\mu} h^1\right). \quad (8)$$

Таким образом, содержательная величина $\delta T = \frac{1}{\mu} s' = -\frac{S}{\mu} (h^0 + \lambda h^1)$ существенно зависит от масштаба μ .

Конечно, от μ зависит и λ :

$$\lambda = - \left\{ \sum_n h_n^0 h_n^1 + \frac{h^0 h^1}{\mu^2} \right\} / \left\{ \sum_n (h_n^1)^2 + \frac{1}{\mu^2} (h^1)^2 \right\},$$

поэтому зависимость δT от μ достаточно сложная, но некоторые качественные выводы можно делать и из формул (8). Используем их для определения μ , руководствуясь следующим принципом: так как $\delta u(\cdot)$ и δT являются «равноправными» компонентами управления, следует стремиться к тому, чтобы вызываемые ими вариации функционалов были величинами одного порядка. Вычислим, в частности, $\delta F_0 = \sum_n s_n h_n^0 + s h^0$ и потребуем равенства (по порядку величины)

$$\left| \sum_n s_n h_n^0 \right| \approx \left| \sum S(h_n^0 + \lambda h_n^1) h_n^0 \right| \approx s' \frac{1}{\mu} h^0 = \frac{1}{\mu^2} S(h^0 + \lambda h^1) h^0.$$

Используя подобные соотношения, выберем масштаб μ по усредненной формуле

$$\mu^2 = \frac{\sum_{i=0}^m (h^i)^2}{\sum_{i=0}^m \sum_n (h_n^i)^2}.$$

Эту величину μ будем считать естественной. Если вычисления проводить с масштабом μ , существенно большим естественного, мы столкнемся с тем, что практически $\delta T \simeq 0$, и процесс минимизации приведет к управлению, оптимальному при постоянном, в сущности, T , заданном начальным приближением. Разумеется, после этого ситуация изменится, уже нельзя пренебрегать в формулах (8) влиянием λ и делать выводы о практически неизменном T . Однако одновременно дело осложняется тем, что вариация

$\delta u(\cdot)$, не влияя на δF_0 , порождает погрешности $O(\|\delta u\|^2)$ в условиях, погашение которых и становится основным назначением вариации $\delta u(\cdot)$, δT . Во всяком случае, процесс минимизации по T существенно осложняется.

§ 50. Модифицированная функция Лагранжа

Стремление свести задачу на условный экстремум к задаче безусловной оптимизации всегда было одной из ведущих тенденций теории экстремальных задач. Это сведение осуществляется фундаментальной теоремой Куна—Таккера. Задача

$$\min_{u \in U} f^0(u) \quad \text{при условиях} \quad f^i(u) = 0, \quad i = 1, \dots, m \quad (1)$$

при некоторых условиях эквивалентна суперпозиции безусловных задач

$$\max_g F(g), \quad \text{где} \quad F(g) = \min_{u \in U} \Lambda(u, g), \quad (2)$$

а функция Лагранжа имеет вид $\Lambda(u, g) = f^0(u) + \sum_{i=1}^m g_i f^i(u)$. Если какое-то условие имеет форму неравенства: $f^i(u) \leq 0$, появляется соответствующее ограничение множителя Лагранжа $g^i \geq 0$. Попытки использовать редукцию (2) в расчетах были не очень успешными: либо в реальных задачах не было выпуклости f^0 , либо, если даже выпуклость была гарантирована, радиус кривизны поверхности $\{f^0(u_g), \dots, f^m(u_g)\}$, где $u_g = \arg \min_{u \in U} \Lambda(u, g)$ (множи-

тели Лагранжа g задают параметризацию этой поверхности) оказывался слишком большим, а сходимость — очень медленной; особенно плохо выполнялись условия $f^i(u) = 0$. В дальнейшем в центре внимания оказался другой способ, использующий сведение (1) к задаче

$$\min_{u \in U} f^*(u), \quad \text{где} \quad f^*(u) = f^0(u) + \sum_{i=1}^m \alpha_i [f^i(u)]^2, \quad (3)$$

где α_i — большие числа, коэффициенты штрафа. Если условие имеет вид $f^i(u) \leq 0$, в (3) берется не $[f^i(u)]^2$, а $[\max(f^i, 0)]^2$. Задача (3) не эквивалентна (1), но аппроксимирует ее с тем большей точностью, чем больше α . Конструкция (3) в некотором роде замечательна: это универсальное средство, позволяющее справиться с любыми трудностями. Если в задаче встречаются какие-то сложные ограничения (условия) и не очень ясно, как их обеспечить, всегда можно сказать: эти условия могут быть учтены методом штрафных функций. После того как конструкция (3) получила достаточно полное математическое обоснование (доказательства теорем о сходимости при $\alpha \rightarrow \infty$ решения (3) к решению (1) были опубликованы многими авторами независимо), начались и попытки

использовать (3) в практических расчетах, и как только они вышли за пределы тестов, отношение к методу штрафных функций стало более скептическим. Оказалось, что при больших α функция $f^*(u)$ становится очень негладкой, трудной для алгоритмов минимизации, а полученные результаты не очень надежны и часто сомнительны. Если же α недостаточно велики, сказывается неэквивалентность задач (1) и (3). Трудной проблемой оказался подбор коэффициентов α_i , особенно при большом числе ограничений m . В последние годы появились работы, в которых объединяются конструкции (2) и (3) в так называемой *модифицированной функции Лагранжа* (М. Ф. Л.):

$$M(u, \lambda, \alpha) = f^0(u) + \sum_{i=1}^m g_i f^i(u) + \frac{\alpha}{2} \sum_{i=1}^m [f^i(u)]^2, \quad (4)$$

причем коэффициент штрафа α уже не обязательно должен быть велик, слагаемое $\alpha \sum_{i=1}^m [f^i(u)]^2$ имеет целью лишь обеспечить выпуклость вниз упомянутой поверхности (быть может, только в окрестности решения) с не очень большим радиусом кривизны. Первые опыты применения М. Ф. Л. оказались обнадеживающими, во всяком случае, они продемонстрировали определенный прогресс по сравнению с методами, использующими (2) или (3).

Алгоритм решения задачи (1) с помощью конструкции (4) является «двухступенчатым». Пусть имеется некоторое приближение $\{u, g, \alpha\}$. Сначала решается задача

$$\min_{u \in U} M(u, g, \alpha). \quad (5)$$

Используется какой-нибудь алгоритм безусловной оптимизации, например, метод градиента, усиленный привлечением идей метода сопряженных градиентов. Точка u используется, как начальная в этом процессе спуска. После этого в новой точке $\{u, g, \alpha\}$ делается пересчет множителей Лагранжа:

$$g_i := g_i + \alpha f^i(u). \quad (6)$$

Из операций (5), (6) и состоит основной итерационный цикл. Теоретическими исследованиями доказана сходимость метода (аккуратные формулировки теорем см. в [10], [11], [25]) со скоростью геометрической прогрессии, знаменатель которой может быть сделан сколь угодно малым за счет достаточно большого α . Однако этот результат не очень полезен в практической работе: дело в том, что увеличение α повышает эффективность «внешнего» итерационного цикла (5), (6), но одновременно отрицательно сказывается на эффективности «внутреннего» итерационного процесса, с помощью которого решается задача (5). В частности,

при очень больших значениях α М. Ф. Л. (4) переходит в конструкцию (3) метода штрафных функций со всеми присущими ей недостатками. Поэтому, хотя сходимость доказывается для любого α , подбор «оптимального» α и в (4) играет большую, можно сказать, определяющую роль. В настоящее время опыт решения задач с помощью М. Ф. Л. еще не очень велик и недостаточно освещен в литературе. Наиболее продвинутыми являются приложения М. Ф. Л. к решению задач линейного программирования. Опыт вычислений освещен в ротапринтных публикациях [10], [11], [74]. Мы разберем этот случай подробнее, так как в задачах линейного программирования заведомо выполнены условия теорем сходимости, а схема метода, основанного на М. Ф. Л., обнаруживает определенное сходство со схемой итерационного метода § 48.

М о д и ф и ц и р о в а н н а я ф у н к ц и я Л а г р а н ж а
в з а д а ч а х л и н е й н о г о п р о г р а м м и р о в а н и я.
Здесь нам будет удобно использовать для задачи линейного программирования следующую компактную запись: найти $\min_s (h^0, s)$ при условиях

$$X + Hs = 0, \quad s^- \leq s \leq s, \quad (7)$$

где $s = \{s_1, s_2, \dots, s_N\}$; X — заданный вектор размерности m ; h^0, s^-, s^+ — заданные векторы размерности N ; H — заданная матрица $N \rightarrow m$. Модифицированная функция Лагранжа для задачи (7) имеет вид

$$M(s, g, \alpha) \equiv (h^0, s) + (g, X + Hs) + \frac{\alpha}{2} \|X + Hs\|^2. \quad (8)$$

Решение задачи (7) осуществляется чередованием следующих операций:

I. При фиксированных g, α находится $\min_s M(s, g, \alpha)$. Разумеется, речь идет о приближенном решении, определяемом заданным числом K итераций того же типа, например, которые используются в алгоритме § 48. Ведь и в данном случае речь идет о нахождении минимума квадратичной формы с двусторонними ограничениями переменных s_n .

II. Делается один шаг в целях максимизации M по двойственным переменным g :

$$g := g + \alpha x, \quad x = X + Hs. \quad (9)$$

Первая операция преследует те же цели, которые в алгоритме § 48 осуществляют блоки 2 и 3. Только там предварительно разделены ресурсы (величины s_n): часть из них ($n \notin N_*$) целиком определяется задачей $\min \{(h^0, s) + (g, X + Hs)\}$, другая часть ($n \in N_*$), которая в силу условия $H_n^0 = h_n^0 + (h_n, g) \approx 0$ мало что

дает для минимизации $\{(h^0, s) + (g, X + Hs)\}$, целиком определяется интересами минимизации $\|X + Hs\|^2$. В (8) эти цели совмещены, причем выбор α определяет (неявным и трудно контролируемым способом) приоритет того или другого фактора. Операция II — пересчет g по формуле (9) — аналогична используемой и в методе § 48. Основная разница в том, что в § 48 параметр α определяется четкой задачей

$$\max_{\alpha} \min_s \{(h^0, s) + (g + \alpha x, X + Hs)\},$$

здесь же параметр α назначается. Обоснованию формулы (9) предположим некоторые простые факты.

Л е м м а 1. *Обозначим через Λ минимальное значение формы (h^0, s) в решении задачи (7). Тогда при любом g имеет место оценка $\min M(s, g, \alpha) \leq \Lambda$.*

Доказательство. Пусть s^* — решение задачи (7), т. е. $(h^0, s^*) = \Lambda$, $X + Hs^* = 0$. Тогда

$$\begin{aligned} \min_s M(s, g, \alpha) &\leq M(s^*, g, \alpha) = \\ &= (h^0, s^*) + (g, X + Hs^*) + \frac{\alpha}{2} \|X + Hs^*\|^2 = \Lambda. \end{aligned}$$

Теорема 1. *Пусть при некоторых g и α $\min_s M(s, g, \alpha)$ достигается в точке s' , для которой $X + Hs' = 0$. Тогда s' есть решение задачи (7).*

Доказательство. В этом случае (h^0, s') определяет оценку Λ как сверху, так, в силу леммы 1, и снизу. Следовательно $(h^0, s') = \Lambda$.

Лемма 2. *Пусть точка s^* минимизирует значение $M(s, g, \alpha)$. Тогда переход к вектору $g^* = g + \alpha x^*$, где $x^* = X + Hs^*$, приводит к росту функции M .*

В самом деле,

$$\begin{aligned} M(s^*, g + \alpha x^*, \alpha) &= (h^0, s^*) + (g + \alpha x^*, X + Hs^*) + \frac{\alpha}{2} \|X + Hs^*\|^2 = \\ &= (h^0, s^*) + (g, X + Hs^*) + \frac{\alpha}{2} \|X + Hs^*\|^2 + \alpha (x^*, X + Hs^*). \end{aligned}$$

Итак, $M(s^*, g + \alpha x^*, \alpha) = M(s^*, g, \alpha) + \alpha \|x^*\|^2$. Этот факт и служит оправданием формулы (9). Заметим, однако, что это не совсем то, что нужно. Более важным было бы доказательство соотношения

$$\min_s M(s, g + \alpha x^*, \alpha) \geq \min_s M(s, g, \alpha).$$

Однако оно, вообще говоря, места не имеет. Тем не менее, использование формулы (9) обосновано доказательством сходимости процесса в целом (см. [63], [64], [25]). Другим основанием формулы

(9) служит следующее рассуждение. Рассмотрим для простоты задачу без ограничений на переменные s . Тогда в точке $s^* = \arg \min_s M(s, g, \alpha)$ можно вычислить производную $\partial M/\partial s$ и приравнять нулю. Получим

$$\frac{\partial M}{\partial s} = h^0 + H^*g + \alpha H^*(X + Hs^*) = h^0 + H^*(g + \alpha x^*),$$

$$x^* = X + Hs^*.$$

Если бы вектор \bar{g} обеспечивал при нахождении $\min_s M(s, \bar{g}, \alpha)$ выполнение условия $X + Hs^* = 0$, мы имели бы соотношение $h^0 + H^*\bar{g} = 0$. Это тоже служит основанием для того, чтобы новое значение g брать в виде $g + \alpha x^*$. Можно было бы попытаться, рассмотрев однопараметрическую конструкцию $g(\beta) = g + \beta x^*$, определить параметр β решением естественной задачи

$$\max_{\beta} \min_s \left\{ (h^0, s) + (g + \beta x^*, X + Hs) + \frac{\alpha}{2} \|X + Hs\|^2 \right\}. \quad (10)$$

Однако решение этой задачи было бы слишком сложным. В методе § 48 аналогичная задача для определения параметра α : $\max_{\alpha} \min_{x \in P} (g + \alpha y, x)$ могла быть решена благодаря тому, что явно выписывается решение задачи $\min_{x \in P} (g, x)$ при любом g .

Опыты вычислений. Первые примеры решения задач линейного программирования с помощью М. Ф. Л. были проведены в [74], затем последовали аналогичные работы [10], [11], (к сожалению, эти публикации практически недоступны широкому кругу читателей). Реализация алгоритма привела к необходимости более точно определить некоторые детали вычислительной технологии. В частности, в [74] для ускорения сходимости формула (9) была преобразована в $g := g + \gamma \alpha x^*$, где $\gamma > 1$ — некоторый множитель, подбираемый экспериментально. В [10], [11] рекомендуется следующая техника:

I*. Внутренние итерации решения задачи $\min M(s, g, \alpha)$ ведутся до выполнения критерия $\|\partial M/\partial s\| \leq 0,2 \|X + Hs\|$, причем при вычислении $\|\partial M/\partial s\|$ учитываются только те компоненты $\partial M/\partial s_n$, для которых $s_n^- < s_n < s_n^+$.

II*. Если цикл внутренних итераций не привел к уменьшению $\|X + Hs\|$ в 10 раз, параметр α удваивается. В противном случае он не меняется. В работах [10], [11] утверждается, что алгоритмы, использующие М. Ф. Л., показали явное преимущество перед алгоритмами, использующими штрафные функции, и перед симплекс-методом, позволяя получить решение с нужной точностью за существенно меньшее время. Приведены и соответствующие числовые данные. К сожалению, все эти работы представляют

собой достаточно распространенные примеры неправомерных выводов из полученных результатов. В этих работах (к ним можно добавить и [61]) полученное приближенное решение характеризуется двумя числами: машинным временем и величиной невязки $\|X+Hs\|$ в конце расчета. Сравнение только этих характеристик служит основанием для заключений об эффективности алгоритма. Между тем есть еще одна существенная характеристика, аналогичная используемой в § 48 величине η . Она характеризует точность решения по величине минимизируемой формы (h^0, s) . Только в том случае, когда метод позволил за меньшее время получить меньшие значения и $\|X+Hs\|$, и (h^0, s) , чем какой-то другой, можно говорить о его явном преимуществе. Сравнение же лишь по величине $\|X+Hs\|$ не дает оснований для каких-либо выводов. В упомянутых работах никакого сравнения по величине (h^0, s) не делается, и связанные с этим вопросы не обсуждаются, соответствующие числовые данные не приводятся. Таким образом, опубликованные в [74], [10], [11] данные позволяют лишь утверждать, что был получен явный прогресс в решении задачи: найти допустимый план задачи линейного программирования (3). Разумеется, не исключено, что на самом деле были получены неплохие результаты и в решении задачи (3), однако опубликованные данные не позволяют судить об этом. Некоторые основания трактовать результаты с малыми значениями $\|X+Hs\|$ как решение задачи (3) дает теорема 1. Однако ее применение требует уверенности в том, что можно пренебречь отличием найденной точки s от точной точки минимума $M(s, g, \alpha)$ и величиной $\|X+Hs\|$. К сожалению, при больших значениях α быстро достигаются очень малые значения $\|X+Hs\|$, однако минимизация $M(s, g, \alpha)$ крайне затруднена, протекает очень неэффективно, и легко принять медленную сходимость за достижение минимума. Поэтому нужно иметь какие-то объективные критерии, по которым данное α можно считать большим или малым. Автор может предположить только следующий способ: получив малые значения $\|X+Hs\|$ (а что такое малое значение $\|X+Hs\|$ обычно в содержательных задачах известно), следует оценить разность

$$|M(s^*, g, \alpha) - \min_s M(s, g, 0)|. \quad (11)$$

Величина $\min_s M(s, g, 0) = \min_s \{(h^0, s) + (g, X+Hs)\}$ легко вычисляется точно (см. формулу для $F(g)$ в § 48) и представляет собой точную оценку снизу для $\Lambda = \min (h^0, s)$, тогда как $M(s^*, g, \alpha)$ является хорошей (при $\|X+Hs\|=0$ — точной) оценкой Λ сверху. По существу, это то же самое, что и используемая нами в § 48 величина η . Кстати, читатель легко поймет, что изложенная выше техника регулирования числа внутренних итераций

и величина 0,2 в пункте I*, предложенная в [11], явно имеет в виду только получение малых значений $\|X + Hs\|$ и совершенно не учитывает необходимость получения соответственно малой величины (14). В какой-то мере можно учесть и интересы величины η , задавая малые стартовые значения α , но это способ очень ненадежный. Автором были проведены некоторые эксперименты по решению задачи линейного программирования с помощью М. Ф. Л. Расчеты проводились с задачей, матрица которой h_n^i заполнялась случайными числами; $s_n^- = 0$, $s_n^+ = \infty$, X^i также были случайными. В первом примере решалась задача с $m=10$, $N=20$. Параметры α и K (число внутренних итераций в минимизации $M(s, g, \alpha)$ в процессе покоординатного спуска) были назначены из сравнения характерных величины X , h , никакого специального их подбора не производилось. Первая же попытка оказалась удачной: за 70 итераций (т. е. пересчетов вектора g) было получено решение с $\eta \approx 0,7\%$, $\|X + Hs\| \approx 0,004 \|X\|$. В действительности при более удачном выборе K и α можно было бы получить и лучшие результаты. Следующий эксперимент проводился на задаче с $m=30$, $N=50$. В этом случае уже пришлось приложить определенные усилия по подбору параметров и усложнению самого алгоритма, прежде чем удалось получить аналогичные по точности результаты. Была проведена серия попыток решения одной и той же задачи, в ходе которой подбирались параметры с целью получить возможно более быструю сходимость. Каждый такой эксперимент занимал на БЭСМ-6 6—12 минут. В процессе отработки алгоритма появились следующие осложнения.

1. Было подобрано число K . При заметно меньших значениях K процесс в целом был неэффективен, при существенно больших — каждая итерация была слишком длительной, что тоже в конце концов приводит к неудовлетворительной эффективности.

2. Был включен некоторый алгоритм пересчета параметра α , предусматривающий как увеличение, так и уменьшение α .

3. Этот алгоритм реагировал на две основные характеристики приближенного решения:

$$\eta = \frac{\left| \min_s M(s, g, \alpha) - \min_s M(s, g, 0) \right|}{\left| \min_s M(s, g, 0) \right|},$$

$$r = \frac{\|X + Hs\|}{\|X\|},$$

и имел целью получить более или менее равномерное стремление обеих характеристик к нулю.

4. Пересчет α после каждой итерации оказался неудачным. Был введен период: π итераций делались при неизменном α , затем α пересчитывалось и т. д.

5. В формулу (9) был введен параметр γ : $g := g + \gamma \alpha x$. Он подбирался экспериментально: из трех испробованных значений $\gamma = 0,1$; $0,3$; 1 наиболее удачным оказалось значение $\gamma = 0,3$. В [74] рекомендуется значение $\gamma > 1$. Это не противоречие, а свидетельство того, что нет универсального наилучшего значения, в разных ситуациях наиболее эффективными оказываются разные значения. Кроме того, в алгоритм были введены и другие параметры, имеющие меньшее влияние. Мы не приводим здесь более точных сведений о значениях параметров, так как они не имеют объективного смысла и существенно связаны с данной конкретной задачей. В другой задаче нужны другие значения. Разумеется, было бы нелепо ставить задачу об отыскании оптимальных значений параметров. Следует разработать алгоритм анализа получающихся в процессе решения характеристик η , r (или еще каких-то), на основании которого можно было бы принимать решение об увеличении или уменьшении того или иного параметра алгоритма. Этого автору сделать не удалось (в работах [74], [11] такая задача и не ставилась). Получив приближенное решение с характеристиками η и r , легко понять, каков желаемый характер дальнейшего хода вычислений также в терминах η , r . Так, если $\eta \gg r$, следует постараться в дальнейшем, быть может, несколько увеличив r — в основном понижать η . Но как придать дальнейшей эволюции r и η нужный характер, изменяя параметры K , α , γ — неясно. Однако результаты экспериментов оставили у автора впечатление, что над этим стоит работать: приложив определенные усилия, можно получить на основе М. Ф. Л. удобный и эффективный алгоритм.

З а м е ч а н и е. Иногда М. Ф. Л. вводится и интерпретируется несколько иначе. В обычной методике штрафных функций с не очень большими коэффициентами штрафа не удается получить хорошее выполнение условий $f'(u) = 0$. Для того чтобы усилить сходимость процесса, не увеличивая коэффициента штрафа, задачу заменяют другой, «сдвигая» требуемые значения $f_i(u)$. Пусть в процессе поиска получена какая-то точка u^* , в которой $f'(u^*) \neq 0$ и в окрестности которой дальнейшая эволюция u происходит слишком медленно. Тогда задача изменяется: вместо условий $f'(u) = 0$ ставятся условия $f'(u) = -\beta f'(u^*)$, где β — некоторый множитель. Тогда функция $f^*(u)$ (3) метода штрафных функций превращается в

$$f^*(u, \beta) \equiv f^0(u) + \sum_{i=1}^m \alpha_i [f^i(u) + \beta f^i(u^*)]^2 = f^0(u) + \\ + \sum_{i=1}^m 2\alpha_i \beta f^i(u) f^i(u^*) + \sum_{i=1}^m \alpha_i [f^i(u)]^2 + \beta^2 \sum_{i=1}^m \alpha_i [f^i(u^*)]^2.$$

Последнее слагаемое в процессе минимизации роли не играет,

а величины $2\alpha_i \beta f^*(u^*)$ отождествляются с компонентами вектора g в М. Ф. Л.

М. Ф. Л. в задачах оптимального управления. Рассмотрим стандартную задачу

$$\min F_0[u(\cdot)]$$

при условиях

$$\dot{x} = f(x, u), \quad u \in U, \quad x(0) = X_0, \quad F_i[u(\cdot)] = 0, \quad i = 1, \dots, m.$$

Функционалы $F_i[u(\cdot)]$ будем считать дифференцируемыми по Фреше. Построение метода приближенного решения этой задачи с помощью М. Ф. Л. приводит к чередующейся последовательности задач.

I. При заданных векторе $g = \{g_1, g_2, \dots, g_m\}$ и параметре α решается задача на безусловный экстремум:

$$\min_{u(\cdot)} F^*[u(\cdot), g, \alpha],$$

$$F^*[u(\cdot), g, \alpha] = F_0[u(\cdot)] + \sum_{i=1}^m g_i F_i[u(\cdot)] + \frac{\alpha}{2} \sum_{i=1}^m F_i^2[u(\cdot)].$$

II. Пересчет g (и, быть может, пересчет параметра α):

$$g_i := g_i + \alpha F_i[u(\cdot)].$$

§ 51. Метод сопряженных градиентов

Метод сопряженных градиентов является итерационным методом нахождения минимума квадратичной формы. Характерная его особенность — конечность: минимум достигается не более, чем за n шагов (n -размерность пространства). Вычислительная схема метода сопряженных градиентов была обобщена на задачи нахождения минимума общих функций, не являющихся квадратичными. Опыт вычислений показал высокую эффективность метода, особенно в ситуациях, когда метод простого спуска по градиенту оказывался практически неработоспособным в силу крайне медленной сходимости. Ниже излагается вычислительная схема метода в случае квадратичной функции, затем будет приведено его формальное обоснование. В заключение будет приведено обобщение вычислительной схемы в случае неквадратичной функции.

Итак, решается следующая задача: найти точку x (в n -мерном евклидовом пространстве), минимизирующую квадратичную функцию

$$f(x) \equiv (a, x) + \frac{1}{2} (Gx, x); \quad (1)$$

G — заданные n -вектор и самосопряженная положительная $n \rightarrow n$ -матрица; при такой G функция $f(x)$ имеет единственную точку минимума.

Приступая к описанию алгоритма, введем полезное обозначение: пусть x, y — два n -вектора. Тогда можно определить линейный оператор ($n \rightarrow n$ -матрицу) формальным выражением

$$A \equiv (x, *)y,$$

где $*$ есть пустое место для аргумента. По определению, действие A на произвольный n -вектор z задается выражением

$$Az \equiv (x, z) \cdot y.$$

Разумеется, нетрудно вычислить элементы a_{ij} матрицы A через компоненты x и y :

$$a_{ij} = x_i y_j.$$

Другими словами, j -я строка A есть вектор x , умноженный на y_j . Легко проверяется формула $A^* = (y, *)x$. Кроме того, мы используем очевидную формулу для градиента:

$$f_x(x) = a + Gx.$$

I. Вычислительная схема метода сопряженных градиентов

Мы опишем стандартный шаг алгоритма. Пусть имеются полученные в предыдущих вычислениях:

x^i , i -е приближение;

g^i — градиент f в точке x^i : $g^i = f_x(x^i) = a + Gx^i$;

H^i — матрица $n \rightarrow n$.

Стандартный шаг состоит в переходе к x^{i+1} , g^{i+1} , H^{i+1} .

1. Вычисляется направление спуска $r^{i+1/2}$;

$$r^{i+1/2} = -H^i g^i.$$

2. Находится шаг $s^{i+1/2}$ спуска по $r^{i+1/2}$, решением задачи

$$\min_s f(x^i + s \cdot r^{i+1/2}),$$

т. е.
$$s^{i+1/2} = -\frac{(a + Gx^i, r^{i+1/2})}{(r^{i+1/2}, Gr^{i+1/2})}.$$

3. Определяются приращение x

$$\Delta x^{i+1/2} = s^{i+1/2} r^{i+1/2}$$

и следующее приближение

$$x^{i+1} = x^i + \Delta x^{i+1/2},$$

4. Вычисляется градиент $f(x)$ в точке x^{i+1} :

$$g^{i+1} = f_x(x^{i+1}) = a + Gx^{i+1}.$$

Отметим следующие используемые в дальнейшем соотношения

$$(\Delta x^{i+1/2}, g^{i+1}) = 0. \quad (2)$$

Это следствие того, что движение по $r^{i+1/2}$ ведется до минимума $f(x + s \cdot r^{i+1/2})$.

5. Вычисление H^{i+1} : находим $\Delta g^{i+1/2} = g^{i+1} - g^i$, далее матрицы (самосопряженные, как легко видеть)

$$A^{i+1/2} = \frac{(\Delta x^{i+1/2}, *) \Delta x^{i+1/2}}{(\Delta x^{i+1/2}, \Delta g^{i+1/2})},$$

$$B^{i+1/2} = - \frac{(H^i \Delta g^{i+1/2}, *) H^i \Delta g^{i+1/2}}{(H^i \Delta g^{i+1/2}, \Delta g^{i+1/2})},$$

и, наконец,

$$H^{i+1} = H^i + A^{i+1/2} + B^{i+1/2}.$$

Этим основной шаг завершен. Заметим, что процесс начинается из произвольной точки x^0 , при $H^0 = E$; все последующие матрицы H^i — самосопряженные.

II. Формальное обоснование сходимости метода

Лемма 1. Для всех $i = 0, 1, \dots$ имеет место равенство

$$H^{i+1} G \Delta x^{i+1/2} = \Delta x^{i+1/2}.$$

Доказательство.

$$1. \quad H^{i+1} G \Delta x^{i+1/2} = (H^i + A^{i+1/2} + B^{i+1/2}) G \Delta x^{i+1/2}.$$

Вычислим отдельно

$$A^{i+1/2} G \Delta x^{i+1/2} = \frac{(\Delta x^{i+1/2}, G \Delta x^{i+1/2}) \Delta x^{i+1/2}}{(\Delta x^{i+1/2}, \Delta g^{i+1/2})} = \Delta x^{i+1/2}.$$

Здесь было использовано важное соотношение

$$\Delta g^{i+1/2} = G \Delta x^{i+1/2},$$

которое следует из определения

$$\Delta g^{i+1/2} = g^{i+1} - g^i = \{a + Gx^{i+1}\} - \{a + Gx^i\} = G(x^{i+1} - x^i).$$

$$2. \quad B^{i+1/2} G \Delta x^{i+1/2} = - \frac{(H^i \Delta g^{i+1/2}, G \Delta x^{i+1/2}) H^i \Delta g^{i+1/2}}{(H^i \Delta g^{i+1/2}, \Delta g^{i+1/2})} =$$

$$= - H^i \Delta g^{i+1/2} = - H^i G \Delta x^{i+1/2}.$$

Доказательство закончено.

Лемма 2. Пусть установлены соотношения:

$$1) \quad (\Delta x^{i+1/2}, G\Delta x^{j+1/2}) = 0, \quad 0 \leq i < j \leq k-1;$$

$$2) \quad H^k G\Delta x^{i+1/2} = \Delta x^{i+1/2}, \quad 0 \leq i \leq k-1.$$

Тогда эти соотношения справедливы соответственно для

$$1) \quad 0 \leq i < j \leq k;$$

$$2) \quad 0 \leq i \leq k.$$

(Заметим, что для начала индукции имеем ($k=1$).

$$1. \quad H^1 G\Delta x^{1/2} = \Delta x^{1/2} \text{ (лемма 1 при } i=0).$$

$$2. \quad (\Delta x^{1/2}, G\Delta x^{1/2}) = 0.)$$

Доказательство. Пусть $j=k \geq 2$, $0 \leq i < k$. Используем соотношение

$$\begin{aligned} x^k &= x^{k-1} + \Delta x^{k-1/2} = x^{k-2} + \Delta x^{k-1/2} + \Delta x^{k-1/2} = \dots \\ &\dots = x^{i+1} + \Delta x^{i+1/2} + \dots + \Delta x^{k-1/2}, \end{aligned}$$

и вычислим

$$\begin{aligned} g^k &= a + Gx^k = a + Gx^{i+1} + G(\Delta x^{i+1/2} + \dots + \Delta x^{k-1/2}) = \\ &= g^{i+1} + G(\Delta x^{i+1/2} + \dots + \Delta x^{k-1/2}). \end{aligned}$$

Теперь вычислим $(\Delta x^{i+1/2}, g^k) = (\Delta x^{i+1/2}, g^{i+1}) + (\Delta x^{i+1/2}, G\Delta x^{i+1/2}) + \dots + (\Delta x^{i+1/2}, G\Delta x^{k-1/2}) = 0$, так как $(\Delta x^{i+1/2}, g^{i+1}) = 0$ — условие движения до минимума $f(x^i + s \cdot r^{i+1/2})$, а $(\Delta x^{i+1/2}, G\Delta x^{j+1/2}) = 0$ для $j = i+1, \dots, k-1$ — по предположению индукции.

Итак, для $i \leq k-1$ установлено

$$(\Delta x^{i+1/2}, g^k) = 0.$$

По предположению индукции для $i \leq k-1$: $H^k G\Delta x^{i+1/2} = \Delta x^{i+1/2}$, следовательно,

$$\begin{aligned} (\Delta x^{i+1/2}, g^k) &= 0 = (H^k G\Delta x^{i+1/2}, g^k) = (\Delta x^{i+1/2}, GH^k g^k) = \\ &= -(\Delta x^{i+1/2}, G\Delta x^{k+1/2})/s^{k+1/2}, \end{aligned}$$

так как имеется соотношение $H^k g^k = -\Delta x^{k+1/2}/s^{k+1/2}$. Итак, $(\Delta x^{i+1/2}, G\Delta x^{k+1/2}) = 0$ для $i \leq k-1$, и первая часть утверждения леммы доказана.

Теперь установим соотношение

$$H^{k+1} G\Delta x^{i+1/2} = \Delta x^{i+1/2} \quad \text{для } i = 0, 1, \dots, k,$$

имея

$$(\Delta x^{i+1/2}, G\Delta x^{k+1/2}) = 0 \quad \text{для } i = 0, 1, \dots, k-1$$

и

$$H^k G\Delta x^{i+1/2} = \Delta x^{i+1/2} \quad \text{для } i = 0, 1, \dots, k-1,$$

причем уже в лемме 1 получено

$$H^{k+1} G\Delta x^{k+1/2} = \Delta x^{k+1/2}.$$

Итак, пусть $i \leq k-1$.

$$H^{k+1}G\Delta x^{i+1/2} = H^k G\Delta x^{i+1/2} + A^{k+1/2}G\Delta x^{i+1/2} + B^{k+1/2}G\Delta x^{1+1/2}.$$

Но

$$\begin{aligned} H^k G\Delta x^{i+1/2} &= \Delta x^{i+1/2}, \\ A^{k+1/2}G\Delta x^{i+1/2} &= \frac{(\Delta x^{k+1/2}, G\Delta x^{i+1/2})}{(\Delta x^{k+1/2}, \Delta g^{k+1/2})} \Delta x^{k+1/2} = 0, \\ B^{k+1/2}G\Delta x^{i+1/2} &= -\frac{(H^k \Delta g^{k+1/2}, G\Delta x^{i+1/2})}{(H^k \Delta g^{k+1/2}, \Delta g^{k+1/2})} H^k \Delta g^{k+1/2}, \end{aligned}$$

так как $(H^k \Delta g^{k+1/2}, G\Delta x^{i+1/2}) = (\Delta g^{k+1/2}, H^k G\Delta x^{i+1/2}) = (\Delta g^{k+1/2}, \Delta x^{i+1/2}) = (G\Delta x^{k+1/2}, \Delta x^{i+1/2}) = 0$, как установлено выше. Здесь было использовано равенство

$$\Delta g^{k+1/2} = g^{k+1} - g^k = \{a + Gx^{k+1}\} - \{a + Gx^k\} = G\Delta x^{k+1/2}.$$

Лемма 2 доказана.

Теперь установим позволяющие начать индукцию утверждения леммы для $k=2$, т. е.

$$\begin{aligned} (\Delta x^{1/2}, G\Delta x^{1/2}) &= 0 & (i=0, j=1), \\ H^2 G\Delta x^{1/2} &= \Delta x^{1/2} & (i=0), \\ H^2 G\Delta x^{1/2} &= \Delta x^{1/2} & (i=1). \end{aligned}$$

Лемма 1 для $i=0$ дает $H^1 G\Delta x^{1/2} = \Delta x^{1/2}$. Вычислим

$$\begin{aligned} (\Delta x^{1/2}, G\Delta x^{1/2}) &= -(\Delta x^{1/2}, s^{1/2}GH^1g^1) = \\ &= -s^{1/2}(H^1G\Delta x^{1/2}, g^1) = -s^{1/2}(\Delta x^{1/2}, g^1) = 0 \end{aligned}$$

(в соответствии с (2) для $i=0$). Найдем

$$H^2 G\Delta x^{1/2} = H^1 G\Delta x^{1/2} + A^{1/2}G\Delta x^{1/2} + B^{1/2}G\Delta x^{1/2}.$$

Но уже установлено $H^1 G\Delta x^{1/2} = \Delta x^{1/2}$, а равенства $A^{1/2}G\Delta x^{1/2} = 0$ и $B^{1/2}G\Delta x^{1/2} = 0$ проверяются так же, как в доказательстве леммы 2. Соотношение же $H^2 G\Delta x^{1/2} = \Delta x^{1/2}$ установлено в лемме 1.

Теперь можно сформулировать окончательный результат.

Теорема 1. Совокупность последовательно вычисляемых n -векторов $\Delta x^{1/2}, \Delta x^{1/2}, \dots, \Delta x^{n-1/2}$ образует ортогональный в G -метрике базис (т. е. $(\Delta x^{i+1/2}, G\Delta x^{j+1/2}) = 0$ при $i \neq j$). В этом базисе установлено n соотношений $H^n G\Delta x^{i+1/2} = \Delta x^{i+1/2}$, $i=0, 1, \dots, n-1$. Следовательно, $H^n G = E$, т. е. $H^n = G^{-1}$. Полученная на n -м шаге процесса точка x^{n+1} есть точка минимума формы $f(x) = (a, x) + \frac{1}{2}(Gx, x)$.

Доказательству подлежит лишь последнее утверждение. В самом деле, последний шаг процесса имеет вид

$$\begin{aligned} x^{n+1} &= x^n + \Delta x^{n+1/2} = x^n + s^{n+1/2}r^{n+1/2} = x^n - s^{n+1/2}H^n g^n = \\ &= x^n - s^{n+1/2}G^{-1}(a + Gx^n) = x^n - s^{n+1/2}x^n - s^{n+1/2}G^{-1}a. \end{aligned}$$

Теперь заметим, что минимум формы $f(x)$ достигается в точке x^* , удовлетворяющей уравнению

$$f_x(x^*) = 0, \quad \text{т. е.} \quad x^* = -G^{-1}a.$$

Поскольку $s^{n+1/2}$ выбирается так, чтобы минимизировать $f(x^n + sr^n)$, то таким значением на последнем шаге процесса будет $s^{n+1/2} = 1$, так как в этом случае $x^{n+1} = -G^{-1}a$ есть точное решение задачи.

Существуют различные вычислительные схемы метода сопряженных градиентов, отличающиеся видом расчетных формул. Будучи формально эквивалентными, эти разные схемы отличаются друг от друга объемом хранимой в процессе вычислений информации, числом операций на стандартный шаг и степенью чувствительности алгоритма к ошибкам округления. Все эти факторы становятся особенно важными при решении задач достаточно высокой размерности.

Кроме того, различные формы алгоритма приводят к различным обобщениям его на задачи с произвольными, не квадратичными функциями. Эти обобщения, разумеется, уже не эквивалентны друг другу и с формальной точки зрения. Ниже мы приведем некоторые формы метода сопряженных градиентов и соответствующие их обобщения на задачи $\min_x f(x)$ с произвольной функцией $f(x)$.

Покажем, как обобщается приведенная выше форма алгоритма.

Алгоритм I^* . Пусть в результате i итераций получены x^i , $g^i = f_x(x^i)$ и матрица $n \rightarrow n$ H^i (в начале процесса x^0 — произвольная точка, $H = E$). Переход к x^{i+1} , g^{i+1} , H^{i+1} осуществляется операциями:

1. $r^{i+1/2} = H^i g^i$.
2. $s^{i+1/2} = \arg \min_s f(x^i + sr^{i+1/2})$.
3. $\Delta x^{i+1/2} = s^{i+1/2} r^{i+1/2}$; $x^{i+1} = x^i + \Delta x^{i+1/2}$.
4. $g^{i+1} = f_x(x^{i+1})$, $\Delta g^{i+1/2} = g^{i+1} - g^i$.
5. $A^{i+1/2} = \frac{(\Delta x^{i+1/2}, *) \Delta x^{i+1/2}}{(\Delta x^{i+1/2}, \Delta g^{i+1/2})}$;
 $B^{i+1/2} = - \frac{(H^i \Delta g^{i+1/2}, *) H^i \Delta g^{i+1/2}}{(H^i \Delta g^{i+1/2}, \Delta g^{i+1/2})}$;
 $H^{i+1} = H^i + A^{i+1/2} + B^{i+1/2}$.

Эта форма алгоритма требует вычисления только градиента функции $f(x)$ и решения (видимо, достаточно точного) одномерной задачи $\min f(x + s \cdot r)$. Кроме того, в процессе решения используется матрица H . Эта форма алгоритма, видимо, не так чувствительна к ошибкам округления, как некоторые другие, более экономные с точки зрения объема памяти и числа операций. Однако

применение этой формы в задачах высокой размерности (возникающих, например, при конечномерной аппроксимации задач в функциональном пространстве) может оказаться затруднительным. Рассмотрим алгоритм с точки зрения объема вычислений:

- 1) вычисление r : $O(n^2)$ арифметических операций;
- 2) вычисление s можно оценить, например, в ≈ 10 вычислений функции f , если применяется алгоритм параболической аппроксимации (число 10, конечно, достаточно условно, но близко к реальному);
- 3) вычисление Δx , x , Δg : $O(n)$ операций;
- 4) одно вычисление $f_x(x)$;
- 5) пересчет H требует $O(n^2)$ арифметических операций (в оценках $O(n)$, $O(n^2)$ коэффициенты близки к 1).

Другие формы метода сопряженных градиентов и их обобщения. ([62]). Алгоритм II. Рассмотрим задачу с квадратичной формой

$$f(x) = \frac{1}{2}(Gx, x) + (a, x); \quad (G = G^*, G > 0).$$

Пусть получены x^i , $r^{i-1/2}$ (в начале x^0 — произвольно, $r^{-1/2} = 0$).

1. $g^i = Gx^i + a \quad (g^i = f_x(x^i)).$
2. $\beta = (Gr^{i-1/2}, g^i) / (Gr^{i-1/2}, r^{i-1/2}) \quad (\text{при } r = 0, \beta = 0).$
3. $r^{i+1/2} = -g^i + \beta \cdot r^{i-1/2}.$
4. $\alpha = -(g^i, r^{i+1/2}) / (Gr^{i+1/2}, r^{i+1/2}).$
5. $x^{i+1} = x^i + \alpha \cdot r^{i+1/2}.$

Заметим, что параметр β определяется условием ортогональности $(r^{i+1/2}, Gr^{i-1/2}) = 0$, а параметр α — условием

$$\min_{\alpha} f(x^i + \alpha \cdot r^{i+1/2}).$$

Отметим, что, в отличие от алгоритма I, в формулы алгоритма II явно входит и квадратичная форма G (в алгоритме I форма G использовалась лишь при вычислении градиента). Это обстоятельство сказывается при обобщении алгоритма на произвольную функцию $f(x)$; при этом возникает необходимость заменить форму G другим подходящим объектом. Естественным аналогом G является матрица вторых производных $f''(x)$.

Алгоритм II*. Пусть получены x^i , $r^{i-1/2}$. Вычисляем:

1. $g^i = f_x(x^i).$
 2. $G = f_{xx}(x^i).$
 3. $\beta = (Gr^{i-1/2}, g^i) / (Gr^{i-1/2}, r^{i-1/2}).$
 4. $r^{i+1/2} = -g^i + \beta r^{i-1/2}.$
 5. $\alpha = \arg \min_{\alpha} f(x^i + \alpha r^{i+1/2}),$
- $$x^{i+1} = x^i + \alpha \cdot r^{i+1/2},$$

Необходимость вычисления $f_{xx}(x)$ во многих задачах (особенно высокой размерности, а именно такие задачи нас особенно интересуют в связи с численным решением задач оптимального управления) препятствует применению алгоритма в такой форме.

Алгоритм III. Пусть получены $x^i, r^{i-1/2}$.

Вычисляем:

1. $g^i = Gx^i + a \quad (g^i = f_x(x^i)).$
2. $\beta = \|g^i\|^2 / \|g^{i-1}\|^2 \quad (\text{при } i=0, \beta=0).$
3. $r^{i+1/2} = -g^i + \beta r^{i-1/2}.$
4. $\alpha = -(g^i, r^{i+1/2}).$
5. $x^{i+1} = x^i + \alpha \cdot r^{i+1/2}.$

Можно показать (см. [62]), что формулы для β в алгоритмах II и III формально (если не учитывать ошибок округления) эквивалентны. Обобщение алгоритма III на произвольную функцию дает

Алгоритм III*. Пусть получены $x^i, r^{i-1/2}$. Вычисляем:

1. $g^i = f_x(x^i).$
2. $\beta = \|g^i\|^2 / \|g^{i-1}\|^2.$
3. $r^{i+1/2} = -g^i + \beta r^{i-1/2}.$
4. $\alpha = \arg \min_{\alpha} f(x^i + \alpha \cdot r^{i+1/2}),$
5. $x^{i+1} = x^i + \alpha r^{i+1/2}.$

Наконец отметим еще одно обобщение алгоритма ([62]).

Алгоритм III**. Он отличается от III* только формулой для параметра β :

$$\beta = (g^i, g^i - g^{i-1}) / \|g^{i-1}\|^2 \quad (\text{при } i=0, \beta=0).$$

Многочисленные модификации алгоритма в случае квадратичных функций $f(x)$ имеют целью ослабить влияние ошибок округлений. Вопрос этот теоретически не разработан и исследуется пока в основном экспериментально.

Для задач оптимального управления, которые при конечно-разностной аппроксимации становятся задачами $\min F[u]$ в пространстве высокой размерности, удобными являются алгоритмы III* и III**, или аналогичные им.

В том виде, в котором они сформулированы выше, алгоритмы применимы к задачам с одним только минимизируемым функционалом $F_0[u(\cdot)]$: нет условий-неравенств $u \in U$, нет дополнительных условий $F_i[u(\cdot)] = 0 (\leq 0)$, $i=1, \dots, m$. Не очень сложно обобщить алгоритм на случай простых ограничений типа $|u| \leq 1$ (покомпонентно). Такие обобщения предложены в [62], использовались они и автором в расчетах [94] (см. § 48).

Не ясно, как включить в схему метода сопряженных градиентов условия $F_i[u(\cdot)] = 0$ (≤ 0), $i=1, 2, \dots, m$. Конечно, здесь, как и в других затруднительных ситуациях, можно сослаться на метод штрафных функций и избавиться от условий $F_i=0$, заменив минимизируемый функционал $F_0[u(\cdot)]$ на составной

$$F[u(\cdot)] = F_0[u(\cdot)] + \sum_{i=1}^m A_i F_i^2[u(\cdot)],$$

введя в задачу большие параметры A со всеми вытекающими из этого отрицательными последствиями.

Метод сопряженных градиентов использовался автором не только в серийных расчетах задач оптимального управления (в качестве одного из блоков решения задачи линейного или квадратичного программирования), но и в методических расчетах в условиях сравнительно высокой размерности. В частности, в § 48 представлены результаты решения задачи линейного программирования итерационным методом, включающим и метод сопряженных градиентов. Видно, что сходимость метода не соответствует теоретическим предсказаниям, что приводит к определенному (и заметному) перерасходу машинного времени. Были проведены и специальные эксперименты по минимизации формы (Bx, Bx) ($G=B^*B$) со случайной матрицей B размером 100×100 . Использовалась схема типа III. Алгоритм не давал нужной точности после 300—400 шагов. Для уменьшения влияния ошибок округления была применена комбинация схем II и III: четыре итерации проводились с вычислением β по схеме III, а каждая пятая — по более громоздкой формуле схемы II. Это привело к улучшению сходимости (выигрыш можно оценить числом ≈ 2), но проблемы не решило.

Как известно, в методе сопряженных градиентов последовательно вычисляемые направления спуска $r^{i-1/2}$ образуют G -ортogonalную систему векторов. Были проведены эксперименты с целью выяснить, как постепенно из-за ошибок округления утрачивается ортогональность. Для этого в процессе решения запоминались некоторые из r , отстоящие от текущего $r^{i+1/2}$ на 20—30 итераций; таких r было три: r' , r'' , r''' . По мере роста номера итерации i эти векторы обновлялись (самый «старый» из них «забывался» и заменялся текущим $r^{i+1/2}$). Вычислялись величины $\gamma' = (Gr^i, r')$, $\gamma'' = (Gr^i, r'')$, $\gamma''' = (Gr^i, r''')$. Результаты показали, что величины γ быстро становятся ненулевыми, хотя очень больших значений не достигают. Обычно они колеблются в пределах 0,01—0,1.

Делались попытки улучшить сходимость, делая добавочную ортогонализацию $r^{i+1/2}$ относительно этих векторов r' , r'' , r''' ,

причем эти векторы брались и такими, как описано выше, и непосредственно предшествующими $r^{k+1/2}$. Заметного улучшения сходимости получить не удалось. Было бы важно разобраться в вопросах влияния ошибок округления на сходимость метода сопряженных градиентов. Не имея хорошей теории этого вопроса, трудно разработать и методы улучшения сходимости. Можно с достаточными основаниями утверждать, что существенным фактором является число обусловленности матрицы G — отношение минимального собственного числа λ_{\min} к максимальному λ_{\max} , причем чем меньше $\lambda_{\min}/\lambda_{\max}$, тем сильнее портится сходимость метода. Поэтому предложенный в [62] переход к базису, в котором G становится возможно более близкой к E , представляется убедительным.

ЛИТЕРАТУРА

1. Абагян А. А., Федоренко Р. П. и др. Some new aspects of the application of the adjoint function and of the perturbation theory in reactor and shielding design. — Женева, 1964. Третья международная конференция по мирному использованию атомной энергии. — Доклад № 364.
2. Артамкин В. Н., Васенкова Г. Н., Отрощенко И. В., Федоренко Р. П. Оптимальный режим остановки реактора. — Атомная энергия, 1964, 17, вып. 3, с. 189—193.
3. Артамкин В. Н., Бабикова Л. П., Федоренко Р. П. Оптимальный режим остановки реактора при проведении краткосрочных работ. — Атомная энергия, 1967, 23, вып. 2, с. 143—145.
4. Аш (Ash M.). Optimal Shutdown Control of Nuclear Reactor. — New York: Academic Press, 1966.
5. Балакришнан (Balacrishnan A. V.). On a new Computing Technique in Optimal Control and its Application to Minimal—Time Flight Profile Optimization — JOTA, 1969, 4, № 1.
6. Беллман Р. Динамическое программирование. — М.: ИЛ, 1963.
7. Беллман, Калаба, Аш (Bellman R., Calaba R., Ash M.). On control of reactor shut-down involving xenon-poisoning. — Nucl. Sc. and Eng., 1959, 6, № 2, p. 152—156.
8. Беллман Р., Дрейфус С. Прикладные задачи динамического программирования. — М.: Наука, 1965.
9. Беллман Р. Процессы регулирования с адаптацией. — М.: Наука, 1964.
10. Белов Е. Н. Алгоритм решения задач линейного программирования. — Программы и алгоритмы, М.: ЦЭМИ, 1973, вып. 47.
11. Белов Е. Н. Алгоритм решения задач квадратичного и линейного программирования. — Программы и алгоритмы, М.: ЦЭМИ, 1974, вып. 57.
12. Болтянский В. Г. Математические методы оптимального управления. — М.: Наука, 1969.
13. Брайсон А. Е., Денхем В. Ф., Дрейфус С. Задачи оптимального управления с ограничениями типа неравенств (I, II). — Ракетная техника и космонавтика (AIAA — Journal). I, 1963, № 11, p. 107—115; II, 1964, № 1, p. 25—34.
14. Будаков Б. М., Беркович Е. М., Соловьева Е. Н. Осциллометрические разностные аппроксимации для задач оптимального управления. — ЖВМ и МФ, 1969, 9, № 3.
15. Бутковский А. Г. Методы управления системами с распределенными параметрами. — М.: Наука, 1975.
16. Вазов В., Форсайт Дж. Разностные методы решения дифференциальных уравнений, с частными производными. — М.: ИЛ, 1963.
17. Ватель И. А., Кононенко А. Ф. Об одной численной схеме решения задач оптимального управления. — ЖВМ и МФ, 1970, 10, № 1, с. 67—37.
18. Величенко В. В. Оптимальное управление составными системами. — ДАН СССР, 1967, 176, № 4, с. 754—756.

19. Величенко В. В. О задаче минимума максимальной перегрузки. — Космические исследования, 1972, X, вып. 5, с. 700—710.
20. Вудкок, Бэбб (Woodcock C., Babb A.). Optimal Reactor Shutdown Programs for Control of Xenon Poisoning. — Trans. Amer. Nucl. Soc., 1965, 8, p. 235.
21. Габасов Р., Кирilloва Ф. М. Принцип максимума для оптимизации систем с запаздыванием. — ДАН СССР, 1970, 194, № 5, с. 995—998.
22. Глестон С., Эдлунд М. Основы теории ядерных реакторов. — М.: ИЛ, 1954.
23. Годунов С. К., Рябенский В. С. Введение в теорию разностных схем. — М.: Физматгиз, 1962.
24. Годунов С. К., Рябенский В. С. Разностные схемы. — М.: Наука, 1977.
25. Гольштейн Е. Г., Третьяков Н. В., Модифицированная функция Лагранжа. — Экономика и матем. методы, 1974, X, вып. 3, с. 568—591.
26. Горлов В. М. О существовании цены игры в задачах преследования. — ЖВМ и МФ, 1972, 12, № 1, с. 78—88.
27. Демьянов В. Ф. К нахождению оптимального управления в задачах автоматического регулирования. — Вестник ЛГУ, 1965, 13, вып. 3, с. 26—35.
28. Дубовицкий А. Я., Милютин А. А. Задачи на экстремум при наличии ограничений. — ЖВМ и МФ, 1965, 5, № 3, с. 395—453.
29. Дубовицкий А. Я., Милютин А. А. Необходимые условия слабого экстремума в задачах оптимального управления со смешанными ограничениями типа неравенств. — ЖВМ и МФ, 1968, 8, № 4, с. 725—779.
30. Дубовицкий А. Я., Рубцов В. А. Линейные быстродействия. — ЖВМ и МФ, 1968, 8, № 5, с. 937.
31. Ермольев Ю. П., Гуленко В. П. Конечно-разностный метод в задачах оптимального управления. — Кибернетика, 1967, № 3.
32. Ивашкин В. Оптимизация космических маневров. — М.: Наука, 1975.
33. Иослович И. О., Борщевский М. З. Некоторые задачи оптимизации стабилизации осесимметричного спутника. — Космические исследования, 1966, вып. 3.
34. Иоффе А. Д., Тихомиров В. М. Теория экстремальных задач. — М.: Наука, 1974.
35. Итеративные методы в теории игр и программирований. — М.: Наука, 1974.
36. Климов А. Д., Федоренко Р. П., Чихладзе И. Л. Решение одной задачи оптимизации импульсного реактора. — М.: ИИМ АН СССР, 1970.
37. Коробов В. И. О сходимости одного варианта метода динамического программирования для задач оптимального управления. — ЖВМ и МФ, 1968, 8, № 2, с. 429—435.
38. Курант (Courant R.). Variational methods for the solution of problems of equilibrium and vibration. — Bull. Amer. Math. Soc., 1943, 49, p. 1—23.
39. Кротов В. Ф., Гурман В. И. Методы и задачи оптимального управления. — М.: Наука, 1973.
40. Крылов И. А., Черноусько Ф. Л. О методе последовательных приближений для решения задач оптимального управления. — ЖВМ и МФ, 1962, 2, № 6, с. 1132—1138.
41. Крылов И. А. Численное решение задачи об оптимальной стабилизации спутника. — ЖВМ и МФ, 1968, 8, № 1.
42. Крылов И. А., Черноусько Ф. Л. Алгоритм метода последовательных приближений для задач оптимального управления. — ЖВМ и МФ, 1972, 12, № 1, с. 14—34.

43. Леончук М. П. О численном решении задач оптимальных процессов с распределенными параметрами. — ЖВМ и МФ, 1964, 4, № 6, с. 1112—1116.
44. Леончук М. П. и др. О численном решении одной задачи оптимального управления ядерными реакторами. — ЖВМ и МФ, 1965, 5, № 3, с. 558—560.
45. Лионс Ж. Л., Латтес Р. Метод квазиобращения и его приложение. — М.: Мир, 1970.
46. Лейтман А. Г. Оптимальное программирование тяги высотных ракет. — В кн.: Исследования оптимальных режимов движения ракет, М.: Оборонгиз, 1959.
47. Лотов А. В. Численный метод исследования непрерывности времени быстрогодействия. — ЖВМ и МФ, 1973, 13, № 5, с. 1315—1318.
48. Лоуден Д. Ф. Оптимальные траектории для космической навигации. — М.: Мир, 1966.
49. Лурье К. А. Оптимальное управление в задачах математической физики. — М.: Наука, 1975.
50. Марчук Г. И. Методы вычислительной математики. — М.: Наука, 1977.
51. Мельц И. О. Применение метода динамического программирования. — Автоматика и телемеханика, 1968, № 1, с. 79.
52. Миль, Дамулакис, Клотье, Титц (Miele A., Damoulakis J. N., Cloutier J. R., Tietze J. L.). Sequential Gradient-Restoration Algorithm for Optimal Control Problems with Nondifferential Constraints. — JOTA, 1974, 13, № 2.
53. Миль (Miele A.). Recent Advances in Gradient Algorithms for Optimal Control Problems. — JOTA, 1975, 17, № 516.
54. Методы оптимизации с приложениями к механике космического полета. — Сборник под редакцией Лейтмана, М.: Наука, 1965.
55. Моисеев Н. Н. Методы динамического программирования в теории оптимальных управлений. — ЖВМ и МФ, I, 1964, 4, № 3; II, 1965, 5, № 1.
56. Моисеев Н. Н. Численные методы теории оптимального управления, использующие вариации в пространстве состояний. — Кибернетика, 1966, 5, № 3, 1—23.
57. Моисеев Н. Н. Численные методы в теории оптимальных систем. — М.: Наука, 1971.
58. Нойштадт (Neustadt L. W.) Synthesis of time-optimal control systems. — J. Math. Anal. Appl., 1960, 1, p. 484—492.
59. Орлов В. В., Федоренко Р. П. и др. Оптимизация физических характеристик защиты от излучения. — В сб.: Вопросы физики защиты реакторов, М.: Атомиздат, 1966.
60. Поляк Б. Т. О некоторых способах ускорения сходимости итерационных методов. — ЖВМ и МФ, 1964, 4, № 5, с. 791—803.
61. Поляк Б. Т. Об одном методе решения задач линейного и квадратичного программирования большого объема. — В сб.: Вычислительные методы и программирование, М.: Изд-во МГУ, 1969, вып. 12.
62. Поляк Б. Т. Метод сопряженных градиентов в задачах на экстремум. — ЖВМ и МФ, 1969, 9, № 4, с. 807—821.
63. Поляк Б. Т., Третьяков Н. В. Об одном итерационном методе линейного программирования и его экономической интерпретации. — Экономика и матем. методы, 1973, VIII, вып. 5, с. 740—751.
64. Поляк Б. Т., Третьяков Н. В. Метод штрафных оценок для задач на условный экстремум. — ЖВМ и МФ, 1973, 13, № 1, с. 34—46.
65. Понтрягин Л. С., Болтянский В. Г., Гамкрелидзе Р. В., Мищенко Е. В. Математическая теория оптимальных процессов. — М.: Физматгиз, 1976.

66. Пшеничный Б. Н. Численный метод расчета оптимального по быстродействию управления для линейных систем. — ЖВМ и МФ, 1964, 4, № 1, с. 52—60.
67. Пшеничный Б. Н., Соболенко Л. А. Ускоренный метод решения задачи линейного быстродействия. — ЖВМ и МФ, 1968, 8, № 6, с. 1345—1351.
68. Пропой А. И. Методы возможных направлений в задачах дискретного оптимального управления. — Автоматика и телемеханика, 1967, № 2, с. 69—79.
69. Пропой А. И. Элементы теории оптимальных дискретных процессов. — М.: Наука, 1973.
70. Розенброк Х., Стори С. Вычислительные методы для инженеров-химиков. — М.: Мир, 1968.
71. Робертс, Смит (Roberts J. J., Smith H. P.). Time Optimal Solution to the Reactivity-Xenon Shutdown Problems. — Nucl. Sc. and Eng., 1975, 22, № 4, p. 470—478.
72. Росточки, Линн (Rostoszy Z., Lynn E.). Optimal Reactor Shutdown Programming for Minimum Xenon Buildup. — Nucl. Sc. and Eng., 1964, 10, № 3.
73. Суворов А. П., Федоренко Р. П. Выбор оптимальных металловодных защит реакторов. — В кн.: Вопросы физики защиты реакторов, М.: Атомиздат, 1969.
74. Сыров Ю. П., Чурквездзе Ш. С. Вопросы оптимизации межотраслевых и межрайонных связей при планировании развития единой народнохозяйственной системы. — Иркутск: Иркутский ин-т народного хозяйства, 1970.
75. Табак, Куо (Tabak D., Kuo B. C.). Application of mathematical programming in the design of optimal control systems. Intern. Journal of Control, 1969, 10, № 5, p. 548—552.
76. Табак, Куо (Tabak D., Kuo B. C.). Optimal Control by Mathematical Programming. — New Jersey: Prentice-Hall Inc. Englewood Cliffs, 1971.
77. Табак Д., Куо В. С. Оптимальное управление и математическое программирование. — М.: Наука, 1975.
78. Тейлор, Смит, Айлиф (Taylor L. W., Smith J., Iliff K. W.). A comparison of minimum time problem for F-104 using Balakrishnan's s-technique. — In: Lect. Notes in Math., № 132. — New York: Springer-Verlag, 1969.
79. Тихонов А. Н. О методах регуляризации задач оптимального управления. — ДАН СССР, 1965, 162, № 4, с. 763.
80. Тихонов А. Н. Об устойчивости задач оптимизации функционалов. — ЖВМ и МФ, 1966, 6, № 4, с. 631.
81. Тихонов А. Н., Галкин В. Я., Зайкин П. Н. О прямых методах решения задач оптимального управления. — ЖВМ и МФ, 1957, 7, № 2, с. 416—424.
82. Шор Н. З. О скорости сходимости метода обобщенного спуска с растяжением пространства. — Кибернетика, 1970, № 2.
83. Шор Н. З., Журбенко Н. Г. Метод минимизации, использующий растяжение пространства. — Кибернетика, 1971, 10, № 3.
84. Шор Н. З., Шабашова Л. П. О решении минимаксных задач методом обобщенного градиента с растяжением пространства. — Кибернетика, 1972, 11, № 1.
85. Шор Н. З. Обобщенные градиентные методы минимизации негладких функций. — Экономика и математ. методы, 1976, XII, вып. 2.
86. Черноуцко Ф. Л., Баничук В. П. Вариационные задачи механики и управления. — М.: Наука, 1973.
87. Федоренко Р. П. Приближенное решение некоторых задач оптимального управления. — ЖВМ и МФ, 1964, 4, № 6.

88. Федоренко Р. П. Опыт итерационного решения задач линейного программирования. — ЖВМ и МФ, 1965, 5, № 4.
89. Федоренко Р. П. Приближенное решение задач оптимального управления. — М.: ИПМ АН СССР, 1968.
90. Федоренко Р. П. Приближенное решение задач линейного программирования высокой размерности. — М.: ИПМ АН СССР, 1968.
91. Федоренко Р. П. Об одной специальной задаче оптимального управления. — ЖВМ и МФ, 1966, 6, № 3, с. 578.
92. Федоренко Р. П. Приближенное решение вариационных задач с недифференцируемыми функционалами. — ЖВМ и МФ, 1971, 11, № 2, с. 348—364.
93. Федоренко Р. П. Итерационное решение задач линейного программирования. — ЖВМ и МФ, 1970, 10, № 4, с. 895—907.
94. Федоренко Р. П. Об итерационном решении задач линейного программирования. — ЖВМ и МФ, 1972, 12, № 2.
95. Федоренко Р. П. О приближенном решении вариационных задач. — ЖВМ и МФ, 1974, 14, № 3, 652—668.
96. Федоренко Р. П. Метод проекции градиента в задачах оптимального управления. — М.: ИПМ АН СССР, 1975, № 5.
97. Филиппов А. Ф. О некоторых вопросах теории оптимального регулирования. — Вестник МГУ, 1959, № 2, с. 25—32.
98. Фресдалъ, Бэбб (Fresdal J., Babb A.) Xenon-135 transient resulting from time-varying shut down of thermal reactors. — Trans. Amer. Nucl. Soc., 1961, 4, p. 316.
99. Энеев Т. М. О применении градиентного метода в задачах оптимального управления. — Космические исследования, 1966, IV, № 5, с. 651.
100. Энеев Т. М. Некоторые вопросы применения метода наискорейшего спуска. — М.: ИПМ АН СССР, 1970, № 17.
101. Янг Л. Лекции по вариационному исчислению и теории оптимального управления. — М.: Мир, 1974

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

- Алгоритм безусловной оптимизации 393
— параболической аппроксимации 393
— условной оптимизации 400
Альтернатива для выпуклого конуса 46
Аппроксимация недифференцируемого функционала 181, 339
— оптимального управления 268, 308, 349, 355
— фазового ограничения 76, 291, 320
- Базисная переменная 419
Базисный вектор 419
Безусловная минимизация 310
Биортогональный базис 420
- Вариация второго порядка 202
— управления конечная 55, 197
— — малая 30, 42, 197
— фазовой траектории 30, 56
— функционала 31, 35, 58, 60, 62, 98
Вектограмма 52, 95
Внебазисная переменная 419
Вторая вариация 203
Выпуклая оболочка 86, 125
Выпуклое тело, множество 369
Выпуклый конус 46
Вырожденная задача линейного программирования 171
Вычислительная технология 173, 210
- Глобальный экстремум 404
- Двухточечные краевые условия 64
Двухшаговый процесс минимизации 407
- Динамическое программирование 122, 305, 389
— — дискретное 387
Дискретная задача управления 386
Дифференциальное включение 86, 88
Дифференциальные управления спуска 383
Дифференцирование по направлению 34
Допустимое управление 46
Допустимый план 419
- Задача быстрогодействия 309
— выпуклого программирования 373
— классического типа 146
— математического программирования 17, 123
— на узкие места 28
— с разрывной правой частью 69
— со свободным временем 68
— строго выпуклого программирования 188, 373
Замыкание множества траекторий 84
- Интервал локализации 408
— управления 24
Искусственный базис 425
- Канторова лестница 91
Касательное многообразие 19, 159
Конечномерная аппроксимация 167, 205, 218
Конечноразностная аппроксимация 229
Конечные связи 157
Континуальная задача линейного программирования 437
Конус запрещенных смещений 72
— допустимых вариаций управления 43, 59, 265

- Конус смещений 44, 60, 63, 243, 261, 266
— убывания функции 414
Коэффициент штрафа 213
Краевые условия общие 25, 65
Критерий качества 26
- Линеаризация 165
Локальная вариация 129
Локальный экстремум 197, 199, 312
Ломаная Эйлера 126
- Матрица влияния 44
Метод бегущей волны 129
— вариаций в пространстве управлений 109
— — в фазовом пространстве 109, 120
— Величенко 315
— второго порядка 201
— Гамкрелидзе 88
— динамического программирования 122, 305
— дробных шагов 129
— золотого сечения 409
— интерпретатора 360
— квазиобращения 357
— Кифера 409
— локальных вариаций 127, 134, 280
— математического программирования 112, 123, 211, 308
— Мельца 162
— Miele 149
— минимальной поправки 148
— Мойсеева 120
— Монте-Карло 405
— наискорейшего спуска 395
— Неймадта—Итона 188, 192
— Ньютона 116, 229, 377, 410
— — модифицированный 379
— обобщенного градиента 412
— первого порядка 201
— поворота опорной плоскости 188
— по координатного спуска 394
— последовательной линеаризации 164, 285
— — минимизации без ограничений 213
— — сверхрелаксации 135
— проектирования градиента 110, 140, 155, 281, 398
— релаксации 135
— случайного спуска 394
— сопряженных направлений 192, 469
— спуска 314, 394
— трубки 133
- Метод тяжелого шарика 406
— условного градиента 148, 223, 400
— штрафных функций 10, 110, 160, 213, 314
— Энеева 111
Минимизирующая последовательность 18, 22, 32, 85, 390, 408, 295, 325, 338, 412
Множество достижимости 44
- Направление спуска 218, 410
Неединственность задачи Коши 117, 236
Некорректность задачи оптимального управления 345
Нормировка задачи 174, 175, 230, 382
- Область достижимости 44, 125, 188, 192, 249
Обобщенный градиент 413
Обратная задача 358
Общие краевые условия 65
Овраг 111, 406
Ограничения общего типа 28, 78, 112
— в фазовом пространстве 27, 112
Одномерный поиск минимума 393
Операторные преобразования градиента 222
Опорная гиперплоскость 189, 370, 372
Опорный вектор 46
Особый режим 236, 313
Отделимость выпуклых тел 371
Ошибка аппроксимации 225, 240, 293
— поиска 213, 293
- Параметр регуляризации 349, 357
— системы 114, 228, 233
Показатель качества 26
Полнота конуса вариации 131, 166
Правило множителей Лагранжа 209, 393, 400
Преобразование Валентайна 111, 161
Принцип максимума 49, 52, 59, 77, 79, 114, 132, 243, 253, 261, 266
— — дискретный 53
Программирование квадратичное 208, 454
— линейное 29, 170, 417, 437
— нелинейное 29
Проектирование 390
— градиента 18, 111, 141
Производная Гато 35, 39, 180
— по направлению 35, 408
— Фреше 21, 30

- Разностная аппроксимация 54, 309
Раскрытие области управления 160
Растяжение пространства 444
Расширенная система 85, 125
Регулирование шага поиска 177, 195, 287, 397, 403, 416
Регуляризация 277, 347, 357
Релейное управление 307, 313
- Сетка в фазовом пространстве 121, 133, 305
Симплекс-метод двойственный 426
— — прямой 419
Склерономные системы 68
Скользящий режим 25, 87, 95, 155, 197
Скрытое решение 95
Согласованная аппроксимация 54, 219
Сопряженное уравнение 32, 234
Сопряженные краевые условия 33
Спуск в пространстве управлений 164
Стационарная траектория 79
— точка метода спуска 395
Строго выпуклая аппроксимация 117, 228, 232
— выпуклое программирование 144
— — тело 144, 369
Сходящаяся в себе последовательность траекторий 84
- Теорема Филиппова 86
Теория регуляризации 357
Терминальная задача 319
Тождество Лагранжа 18, 32, 70, 98, 105
Точки аппроксимации 181, 299, 320, 331
Точность линейного приближения 142, 179, 247, 282
- Траектория 42
—, допустимая вариация 43, 168
—, допустимое управление 44
- Улучшающая вариация 143, 166
Универсальная последовательность шагов спуска 385, 413
Унимодальная функция 408
Управление 20, 24, 61
— в широком смысле слова 61
— формой области 102
Управляемая система 21, 24
Уравнение в вариациях 18, 30, 56, 62, 70, 73, 98, 105
— — — второго порядка 202
— динамического программирования 305, 387
— связи 19
— с запаздыванием 72
— Эйлера 22
Условия входа 158
— неравенства 72
— трансверсальности 64, 67, 261, 266
- Фазовое пространство 24
Фазовые координаты 24
— ограничения 75, 289
Функционал от траектории 26
Функция Беллмана 125, 305
— Гамильтона 48
— Лагранжа 461
— — модифицированная 462
- Шаг спуска 394, 397
- Элементарная операция 121, 126, 128
— —, метод Балакришнана 136

УКАЗАТЕЛЬ ОБОЗНАЧЕНИЙ

- x^i — i -я компонента вектора фазовых переменных.
 u_k — k -я компонента вектора управляющих переменных.
 $f(x, u)$ — правая часть системы уравнений движения управляемой системы.
 $f[t]$ — обозначение для определенной траектории функции:

$$f[t] \equiv f[x(t), u(t)], \quad f_x[t] \equiv f_x[x(t), u(t)] \text{ и т. д.}$$
- Φ — вектор сопряженных переменных.
 A^*, f_x^* — матрицы, сопряженные к A, f_x .
 $\delta x, \delta u, \delta F, \dots$ — единые символы для вариаций x, u, F, \dots соответственно.
 $u(\cdot), x(\cdot)$ — символы функций, рассматриваемых как точки функциональных пространств.
 $u(t), x(t)$ — значения $u(\cdot), x(\cdot)$ в момент времени t .
 $F[u(\cdot)]$ — стандартное обозначение для функционала от $u(\cdot)$.
 $\frac{\partial F[u(\cdot)]}{\partial u(t)}$ — производная Фреше функционала $F[u(\cdot)]$.
 $w(t) \delta u(t), (w(t), \delta u(t))$ — обозначения для скалярных произведений.
 U — область допустимых значений управления u .
 $f(x, u)$ — i -я компонента вектора $f(x, u)$.
 f_x — матрица с элементами $\frac{\partial f_i}{\partial x_j}$.
 $f(x, U)$ — множество точек $f(x, u)$ для всех $u \in U$.
 conv — символ выпуклой оболочки.
 $H(x, \Phi, u)$ — функция Гамильтона.
 $\Gamma(x)=0$ — символическая запись краевых условий.
 $Gx=0$ — символическая запись линейных однородных краевых условий.
 $\text{var } u(\cdot)$ — вариация функции $u(\cdot)$.
 $M \setminus M'$ — разность множеств M и M' .
 K_u — конус допустимых по условию $u(t) + \delta u(t) \in U$ вариаций $\delta u(\cdot)$.
 K_F — конус возмущений значений функционалов.
 $\arg \min_x f(x)$ — точка (или множество точек), в которой достигается $\min_x f(x)$.
 $f(x^*) = \max_x f(x)$ — определение точки x^* как $\arg \max_x f(x)$.
 $\delta U(t)$ — малая окрестность точки $u(t)$.
 ΔF — точное приращение функционала.
 $w_i(t)$ — производная по $u(\cdot)$ функционала $F_i[u(\cdot)]$.
 $\delta(t-t')$ — δ -функция Дирака.
 \equiv — обозначение «равно по определению»; слева от знака \equiv помещается определяемый объект, справа — определение.
 ν — номер итерации.
 $x \leq y$ — для векторов x и y означает указанное соотношение для одноименных компонент.
 $x \leq a$ — для вектора x и скаляра a означает указанное соотношение для каждой компоненты x .
 $:=$ — знак операции, заимствованный из языка алгол; означает вычисление величины, стоящей слева, по формуле, написанной справа от знака.
 \approx — знак «примерно равно».
 $a \sim b$ — величина a того же порядка, что и величина b .
 $(0, \dots, 0, 1_i, \dots, 0)$ — вектор, i -я компонента которого равна 1.
 $x = (x^1, x^2, \dots, x^k)$ — вектор, в фигурных скобках — его компоненты.
 $[0, T]$ — интервал изменения независимого аргумента в задаче оптимального управления.

Радий Петрович Федоренко

**ПРИБЛИЖЕННОЕ РЕШЕНИЕ ЗАДАЧ
ОПТИМАЛЬНОГО УПРАВЛЕНИЯ**

(Серия: «Справочная математическая библиотека»)

М, 1978 г., 488 стр. с илл.

Редактор *М. Н. Мушников*

Технический редактор *В. Н. Кондакова*

Корректоры *Е. А. Велицкая, Л. С. Сомова*

ИБ № 11046

Сдано в набор 25.04.78. Подписано к печати 14.09.78. Т-17447.
Бумага 60×90¹/₁₆, тип. № 1. Обыкновенная гарнитура.
Высокая печать. Условн. печ. л. 30,5. Уч.-изд. л. 30,45.
Тираж 12 000 экз. Заказ № 364. Цена книги 1 р. 90 к.

Издательство «Наука»

Главная редакция физико-математической литературы
117071, Москва, В-71, Ленинский проспект, 15

Первая тип. изд-ва «Наука»
199034, Ленинград, В-34, 9-линия, 12

