МИНИСТЕРСТВО ОБОРОНЫ РОССИЙСКОЙ ФЕДЕРАЦИИ

ВОЕННО-КОСМИЧЕСКАЯ АКАДЕМИЯ ИМЕНИ А.Ф. МОЖАЙСКОГО

Б.В. Москвин

ТЕОРИЯ ПРИНЯТИЯ РЕШЕНИЙ

Учебник

Допущено Министерством обороны Российской Федерации в качестве учебника для курсантов высших военно-учебных заведений, обучающихся по направлению "Информатика и вычислительная техника", и для слушателей военных академий, обучающихся по специальности "Управление эксплуатацией вооружения, военной техники и техническим обеспечением войск (сил)"

Санкт-Петербург - 2005

УДК 519.816 (075.8)

М822 **Москвин Б.В.**

Теория принятия решений: Учебник / Б.В. Москвин. – СПб.: ВКА имени А.Ф. Можайского, 2005. – 383 с.

В учебнике рассматриваются математические модели и алгоритмы принятия решений, построенные на основе статических и динамических моделей, а также вопросы принятия решений в сложных военно-технических системах.

Разбираются теоретические основы и вычислительные схемы решения задач математического программирования (линейного, нелинейного и дискретного) и задач оптимального управления, приводятся примеры применения алгоритмов. Рассматриваются факторы, усложняющие процессы принятия решений. Особое внимание здесь уделяется вопросам принятия решений в условиях неопределенности обстановки, в условиях критериальной неопределенности, вызванной противоречивостью требований, предъявляемых к оптимальному решению, принятию решений в военно-технических системах, обладающих сложной иерархической структурой. Приводятся примеры решения военно-прикладных задач различных классов.

Учебник предназначен для самостоятельной подготовки слушателей при изучении дисциплины "Методы и технологии выработки управленческих решений" и курсантов при изучении дисциплины "Теория принятия решений", а также адъюнктов и специалистов, интересующихся вопросами автоматизации принятия решений при управлении сложными военно-техническими системами.

Рецензенты:
Кафедра боевого применения АСУ войсками и связи Военной академии связи;
Санкт-Петербургский институт информатики и автоматизации РАН: доктор технических наук, профессор Соколов Б.В.;
Кафедра автоматизации и обработки информации ВКА имени А.Ф.Можайского:
доктор технических наук, профессор Дмитриев А.К.

Подп. к печ. 21.04.2005	Печ. л. 22,88	Уч.-изд. л. 22,5
Зак. 1512		Свободная цена

Типография ВКА имени А.Ф.Можайского

ОГЛАВЛЕНИЕ

РАЗДЕЛ 3. ПРИНЯТИЕ РЕШЕНИЙ В СЛОЖНЫХ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМАХ

ВВЕДЕНИЕ

Военно-космическая деятельность Российской Федерации, начиная с момента запуска первого искусственного спутника Земли и по настоящее время, тесно связана с решением одной из центральных проблем исследования сложных систем любой материальной природы – проблемой управления космическими средствами. Среди этих средств, в первую очередь, необходимо выделить орбитальные группировки космических аппаратов (КА) различного целевого назначения (навигация, связь, разведка) и соответствующие технические средства, образующие наземные комплексы управления. Основу существующей и перспективной технологии управления перечисленными орбитальными и наземными космическими средствами составляет комплекс операций, связанных с получением, обработкой, анализом траекторной, телеметрической информации о состоянии средств, периодическим формированием и реализацией управляющих воздействий, выдаваемых на борт космического аппарата с использованием технических средств наземного комплекса управления.

Главная особенность и отличительная черта указанных процессов состоит в необходимости оперативной и качественной обработки огромных потоков разнородной информации, производимой с целью выработки рациональных решений по управлению космическими средствами. Развитие и возрастание сложности объектов управления (КА и средств наземного комплекса), повышение требований к качеству управления в связи с возрастанием цены управленческой ошибки, рост технологической сложности процессов управления ведет к повышению интенсивности управленческого труда. Такая тенденция имеет устойчивый характер во всех областях деятельности человека, и она обуславливает необходимость автоматизации управления. Однако, развитие автоматизации даже в передовых отраслях отстает от роста требований к интенсивности обработки возрастающих потоков информации, требований к оперативности и качеству принятия решений - центральной функции управления. В связи с этим во всех отраслях производственной деятельности проявляется устойчивая тенденция роста количества специалистов управленческого характера.

Сложность обстановки, в которой командиру приходится управлять современными военными системами в реальном масштабе времени, требует значительного усиления поддержки процессов подготовки и принятия решений на основе комплексной автоматизации. Основой автоматизированного управления космическими средствами является формализованное описание – математическая модель ситуации принятия решения. В настоящее время имеются значительные успехи в разработке и широком практическом применении математических моделей различных классов, для которых разработаны эффективные алгоритмы оптимизации, позволяющие получать решения в приемлемое для войсковой практики время. Большое разнообразие ситуаций, возникающих при управлении космическими средствами, необходимость оперативного принятия решений, удовлетворяющих разнородным качественным требованиям, определяемым спецификой целевого функционирования военных систем, вызывают необходимость комплексного использования богатого арсенала математических моделей, разработанных в рамках системно-кибернетических исследований. Это, прежде всего, модели математического программирования и оптимального управления, позволяющие учитывать основные особенности функционирования космических средств. В рамках моделей данных классов накоплен значительный опыт разработки достаточно эффективных алгоритмов, позволяющих наиболее полно отражать специфику решаемых задач.

Оптимизация процессов управления военно-техническими системами на основе аналитических статических моделей принятия решений связана с поиском некоторого конечномерного вектора, описывающего решение, который доставляет оптимум (максимум или минимум) целевой функции. Алгоритмы нахождения таких векторов, основанные на использовании методов оптимизации, развиваются в рамках математического программирования - наиболее изученного и широко применяемого на практике направления оптимизации решений. Здесь, прежде всего, различают задачи с непрерывными переменными (компоненты вектора, описывающего решение, действительные числа) и задачи с целочисленными переменными. В зависимости от характера целевой функции и ограничений различают задачи линейные и нелинейные. Одной из первых практических задач данного класса была задача о выборе производственной программы, которая была решена в 1939 г. в работе известного российского математика академика Л.В.Канторовича. Однако наиболее интенсивное развитие математическое программирование получило в период 1955-1970 г.г., и в настоящее время имеются значительные успехи в области создания эффективных алгоритмов решения этих задач, более того, ряд более сложных процедур оптимизации строится на основе использования методов математического программирования.

Совершенствование способов управления космическими средствами, тесно связано с необходимостью создания новых и развития существующих моделей и алгоритмов принятия решений в различных условиях обстановки. В настоящее время наряду с моделями математического программирования все шире в практику управления военно-техническими системами внедряются модели о п т и м а л ь н о г о у п р а в л е н и я. Это обусловлено, прежде всего, тем, что достаточно полное описание ситуаций, возникающих при управлении системами различного рода, требует поиска новых формализмов, используемых при описании решения. И, если в результате проведенного системного анализа обстановки требуется, чтобы в математической модели решение описывалось некоторой функцией, например, функцией времени, то методы математического программирования, как правило, становятся непригодными для поиска наилучшего решения. В таких ситуациях следует использовать методы оптимального управления. Эффективным средством исследования задач оптимального управления является принцип максимума Л.С.Понтрягина, сформулированный в 1956 г. и представляющий собой необходимое условие существования оптимального решения. Принцип максимума представляет собой крупное достижение современной математики, он существенно обобщает и развивает результаты классического вариационного исчисления, и более того, стимулирует последующее интенсивное развитие методов решения экстремальных задач. Многие алгоритмы, основанные на принципе максимума, нашли практическое применение при решении различных задач управления космическими системами и средствами.

Неотъемлемой принадлежностью с л о ж н ы х военно-технических систем является неопределенность и неполнота информации об обстановке, которая используется для принятия решения. Многоцелевой характер функционирования военных систем требует всесторонней оценки качества принимаемых решений на основе использования нескольких показателей (нескольких целевых, критериальных функций). Рост сложности и масштабности целевых задач, решаемых военными системами, приводит к тому, что существующая система разбивается на совокупность подсистем, решающих специализированные задачи, которые имеют информационную, методическую и алгоритмическую общность. Это сопровождается децентрализацией процесса обработки информации и, как следствие этого, децентрализацией принятия решения и возрастанием структурной сложности систем. Рост технологической сложности процессов управления, повышение требований к качеству функционирования военных систем в различных условиях обстановки, обуславливает необходимость разработки соответствующих моделей и алгоритмов, которые используются при автоматизации управления и в достаточной степени отражают указанные особенности сложных военных систем.

Необходимость создания новых и развития существующих методов принятия решений при управлении космическими средствами обусловлена, прежде всего, повышением требований к качественным характеристикам военно-космических систем (боеготовности, оперативности, боевой эффективности), возрастанием количества и разнообразия целевых задач, стоящих перед этими системами. Неопределенность обстановки целевого функционирования систем, обусловленная возможным воздействием противника, неквалифицированными действиями личного состава, быстрой сменой целевых установок командования, требует проводить решение указанных проблем сложности и неопределенности на основе интеграции современных системно-кибернетических методов принятия решений и выработанных в войсковой практике способов подготовки и проведения операций. Автоматизация процессов принятия решений на основе использования методов математического моделирования создает предпосылки *непрерывного совершенствования* технологии и качества управления космическими средствами, которая достигается благодаря возможностям гибкой модернизации соответствующих программных средств. В этой связи *специфическая особенность* профессиональной деятельности военного инженера-системотехника Космических войск Российской Федерации связана с его умениями *комплексного применения* с одной стороны *методов системного анализа* возможностей выполнения поставленных задач, и, с другой стороны, *методов формализованного (математического) описания* возможных вариантов и способов нахождения рациональных решений таких задач.

В настоящем учебнике рассматриваются математические методы и алгоритмы принятия решений на основе статических и динамических моделей, а также вопросы принятия решений в сложных военно-технических системах. Учебник состоит из введения, 16 глав и библиографического списка.

В первой главе определяется роль и место проблемы принятия решений при управлении космическими средствами. Рассматриваются этапы, в результате выполнения которых строится формализованное описание (математическая модель) ситуации принятия решений. Приводится общая постановка задачи принятия решений.

В первом разделе рассматриваются статические модели и алгоритмы принятия решений, построенные на основе методов математического программирования.

Во второй главе приводится постановка задачи математического программирования, рассматриваются ее особенности и условия, которым должны удовлетворять оптимальные решения.

В третьей главе рассматриваются теоретические основы и методы решения наиболее изученных задач математического программирования – задач линейного программирования. Рассматриваются особенности построения алгоритмов, предназначенных для решения общей задачи.

В четвертой главе изучаются задачи нелинейного программирования. Рассматриваются особенности таких задач, различные подходы и алгоритмы, предназначенные для их решения.

В пятой главе приводятся методы решения задач дискретного программирования, получившие наибольшее распространение.

В шестой главе рассматриваются специальные задачи математического программирования, наиболее часто используемые в практических ситуациях. Приводятся алгоритмы, учитывающие специфику таких задач, и примеры их применения.

В седьмой главе учебника собраны примеры решения различных задач математического программирования, иллюстрирующие различные особенности и возможности алгоритмов.

Во втором разделе рассматриваются динамические модели и алгоритмы принятия решений, построенные на основе методов оптимального управления.

В восьмой главе проводится постановка задач оптимального управления, рассматривается классификация и качественный анализ таких задач. Формулируются необходимые условия оптимальности решения в форме принципа максимума.

В девятой главе рассматриваются алгоритмы поиска программы управления, основанные на решении решения специальной краевой задачи.

В десятой главе изучаются вопросы поиска оптимального управления в форме синтеза управления в линейных задачах с непрерывным временем и в многошаговых задачах.

В одиннадцатой главе приводятся примеры практического решения задач оптимального управления на основе принципа максимума Л.С.Понтрягина и на основе принципа оптиальности Р.Беллмана.

В третьем разделе рассматриваются вопросы принятия решений в сложных военно-технических системах.

В двенадцатой главе приводятся различные определения сложной системы, выявляются основные факторы, усложняющие процесс выбора решений при управлении космическими средствами.

В тринадцатой главе приводятся различные способы принятии решений в условиях неопределенного воздействия среды. В зависимости от степени знаний, имеющихся о среде, рассматриваются различные модели и алгоритмы.

В четырнадцатой главе рассматриваются способы принятия решений в ситуациях, когда к оптимальному решению предъявляется несколько противоречивых требований.

В пятнадцатой главе изучаются основы построения алгоритмов принятия решений в сложных системах, обладающих иерархической структурой.

В шестнадцатой главе собраны примеры использования рассмотренных алгоритмов принятия решений в сложных военно-технических системах.

Учебник имеет комплексное предназначение и может использоваться при проведении различных видов занятий, включая лекции, семинары, практические и лабораторные занятия, а также в работе курсантов, слушателей, адъюнктов и специалистов, интересующихся вопросами автоматизации управления.

Автор благодарит Кудряшова А.Н за полезные обсуждения вопросов организации управления космическими средствами и Ефимова А.С. за сделанные замечания по тексту рукописи, а также выражает признательность сотрудникам кафедры Мануйлову Ю.С., Павлову А.Н., Петрошенко А.В., Верзилину Д.Н., Колесникову К.Г. за помощь и поддержку при подготовке учебника к изданию.

1.ПОСТАНОВКА ЗАДАЧ ПОДГОТОВКИ И ПРИНЯТИЯ РЕШЕНИЙ ПРИ УПРАВЛЕНИИ ВОЕННО-ТЕХНИЧЕСКИМИ СИСТЕМАМИ

1.1. Принятие решений в системно-кибернетических исследованиях

Формирование и развитие системных взглядов имеет глубокие корни. На их формирование оказали большое влияние борьба различных концепций на философском уровне, исследования общенаучного уровня, проводимые в рамках конкретных (частных) наук. В настоящее время можно различать две группы факторов, стимулировавших становление и развитие системно-кибернетических исследований. Это внешние и внутренние наукообразующие факторы.

Внешние наукообразующие факторы определяются объективными потребностями общества в развитии научных исследований, обладающих практической ценностью получаемых результатов.

Во все времена одной из основных практических целей научно-технического поиска являлось создание механизмов и машин, облегчающих физический труд человека, составляющий основу общественной производственной деятельности. Первые таких механизмы (рычаг, винт, колесо,…) позволяли значительно увеличить физические возможности человека. Положение дел резко изменилось с началом промышленной революции. Были созданы первые паровые машины, позволяющие использовать технические устройства в качестве преобразователя энергии пара, т.е. исключить человека, как источника силы. В дальнейшем были разработаны двигатели внутреннего сгорания топлива, двигатели, основанные на преобразовании атомной энергии. Началось интенсивное сближение и взаимно-стимулирующее развитие науки и техники, которое в последующем получило название научно-технического прогресса.

Наряду с направлением научно-технического прогресса, связанным с повышением производительности физического труда человека, развивалось и другое направление, связанное с автоматизацией процессов не физического, но умственного труда, связанное с передачей машине мыслительных функций человека. Это направление качественно отличается от первого, так как для передачи машине мыслительных операций требуется обязательное предварительное формализованное описание этих операций в символах некоторой формальной теории. Такой универсальной, формальной основой решения разнообразных задач стала математика. В связи с созданием электронных вычислительных машин, обладающих значительной памятью и высоким быстродействием, стала возможной реализация высокоэффективных математических методов и алгоритмов. Разработка таких прикладных математических теорий стала интенсивно развиваться в рамках системно-кибернетических исследований.

Внутренние наукообразующие факторы определяются потребностями самой науки в развитии исследований, способствующих получению новых результатов.

Система научных знаний, начиная с глубокой древности и до 17-18 веков, развивалась в основном как единая философская наука, носящая энциклопедический характер. Однако по мере накопления знаний и расширения сферы научных исследований, обусловленного требованиями общественной практики, в развитии науки все более отчетливо стали проявляться тенденции к дифференциации знаний, выражающиеся, в первую очередь, в увеличении количества отдельных наук и в ослаблении связей между ними. Процесс дифференциации объективно отражает внутреннюю динамику развития системы научных знаний, когда объем научной информации постоянно нарастает. Вместе с тем, он имеет существенные негативные последствия, связанные, прежде всего, с возникновением "барьеров" специализации, затрудняющих взаимопонимание между специалистами и, главное, обмен опытом и результатами выполненных исследований. Анализ внутреннего развития науки показывает, что для преодоления указанных негативных последствий дифференциации научных знаний могут быть использованы два пути.

Первый путь связан с разработкой междисциплинарных направлений, обеспечивающих необходимую концептуальную и методологическую взаимосвязь между различными дисциплинами. Этот путь приводит к возникновению новых, "гибридных" наук, таких, например, как физическая химия, биофизика, и т.п. Однако, подобное развитие системы научных знаний приводит к дальнейшему дроблению областей научного поиска и позволяет лишь частично решить проблему дифференциации.

Второй путь заключается в разработке комплекса обобщающих взглядов, понятий и концепций, на основе которых оказывается возможным объединение широкого круга, на первый взгляд, разрозненных областей исследований. Наиболее ярким современным примером указанного пути является интеграция знаний на основе системного подхода и введения таких обобщающих понятий, как "система", "среда", "управление" и ряд других, позволяющих выявлять и исследовать строение и закономерности функционирования различных объектов реального мира, инвариантные по отношению к их материальной природе. Теоретическое направление в современной науке, связанное с изложенной концепцией, принято называть в настоящее время с и с т е м н о - к и б е р н е т и ч е с к и м и исследованиями.

Предметом системно-кибернетических исследований является изучение структуры и общих закономерностей функционирования систем любой материальной природы, исследование свойств этих систем и разработка методов и алгоритмов управления ими на основе построения и исследования соответствующих концептуальных и математических моделей. В рамках системно-кибернетических исследований уточняются понятия системы и среды, вводятся такие фундаментальные понятия, как состояние системы, управляющая и возмущающая среда, рассматривается общая морфология системно-кибернетических проблем. Рассматриваются основные математические модели систем, строгие постановки проблем выбора и методы их решения. Таким образом, системно-кибернетические исследования представляют собой фундаментальное методологическое направление в современной науке.

В основе системно-кибернетических исследований лежит системный подход.

С и с т е м н ы й п о д х о д - общая методология исследования объектов природы, общества, науки и техники как сложных систем.

Системный подход оказался весьма конструктивным при исследовании разнообразных и весьма сложных реальных систем, при разработке методов и алгоритмов управления такими системами. Практическая эффективность системного подхода объясняется тем, что он, с одной стороны, позволяет достаточно всесторонне понять суть исследуемых явлений и процессов и построить их концептуальную модель в терминах соответствующей теории. С другой стороны, присущий системному подходу абстрактный уровень мышления хорошо согласуется с концепциями и конструкциями современной математики, что, в свою очередь, обеспечивает строгость и корректность построения соответствующих математических моделей систем и возможность использования богатейшего, современного математического инструментария при исследовании реальных систем и управлении ими.

В рамках системного подхода вводятся такие обобщающие понятия, как система и среда.

С и с т е м а - целостное образование, состоящее из взаимосвязанных (взаимодействующих) компонентов (элементов, частей) и обладающее свойствами, не сводимыми к свойствам этих компонент и не выводимыми из них.

С р е д а - совокупность элементов окружающего мира, не входящих в систему, но оказывающих на нее существенное влияние. Открытая система - система, взаимодействующая со средой.

В качестве целей системно-кибернетического исследования различают:

- анализ свойств системы (устойчивости, эффективности, наблюдаемости, управляемости,...);
- разработка управляющих воздействий на систему, предназначенных для достижения целей функционирования системы.

Проблемы принятия решений связаны именно со второй группой системно-кибернетических исследований - с управлением системой.

У п р а в л е н и е - процесс целенаправленного изменения состояния (выхода) системы, осуществляемый путем воздействия на нее (управляющего воздействия).

С о с т о я н и е - совокупность параметров (характеристик), отражающих наиболее существенные свойства системы и определяющих ее поведение (функционирование).

Ц е л ь, рассматриваемая в аспекте функционирования системы, есть желаемое состояние системы, достигаемое или поддерживаемое управляющими воздействиями (управлениями).

Для разработки и реализации управляющих воздействий (управлений) создается специализированная система - система управления, которая воздействует на объект управления - исследуемую военно-техническую систему. В результате управляющего воздействия на систему изменяется ее состояние и выходные характеристики, анализ которых служит основой для выработки новых управляющих воздействий. В зависимости от степени участия человека в управлении различают:

- системы автоматического управления (САУ) - разрабатываются законы воздействия на систему, позволяющие осуществлять управление без участия человека;
- автоматизированные системы управления (АСУ) - человек (оператор) непосредственно участвует в процессе управления.

Последний вид систем управления (АСУ) нашел широкое применение в практике эксплуатации сложных военно-технических систем (ВТС). Весь процесс управления в таких системах связан с выполнением некоторой совокупности функций управления, которые реализуются в определенной последовательности в соответствии с *технологией* управления:

- сбор информации об обстановке (состояние и выход военно-технической системы, состояние среды,...) - (С);
- анализ состояния системы и среды по результатам сбора информации - (А);
- прогноз состояния системы и среды на интервале управления - (П);
- выбор (принятие) решения по управлению - (ВР);
- доведение решения до исполнителей - (Д);
- реализация принятого решения и контроль за исполнением решения - (Р).

Последовательность функций управления представляет собой технологический цикл управления, который можно характеризовать схемой.

Рис. 1.1

Здесь

$t_с$ - момент времени, характеризующий начало сбора информации;

$t_а$ - время начала анализа информации;

$t_п$ - время начала прогноза обстановки и состояния системы;

$t_р$ - время начала выработки решения;

$t_д$ - время начала доведения до исполнителей принятого решения;

$t_х$ - время конца доведения решения;

$t_о$ - время начала реализации принятого решения;

t_f - время конца реализации принятого решения.

Тогда

$t_f - t_с$ - длительность цикла управления;

$(t_о, t_f]$ - интервал управления;

Важнейшей функцией управления является функция выбора (принятия) решения (ВР), которая составляет основу разработки управляющих воздействий на систему с целью достижения ее целей. Под решением будем понимать следующее

Р е ш е н и е - управляющее воздействие, направленное на достижение целей системы.

Таким образом, основой управления сложными военно-техническими системами является решение, анализ и выбор которого производится с использованием формальных математических моделей.

Т е о р и я п р и н я т и я р е ш е н и й - научная дисциплина, изучающая методы формализации (математического описания) ситуаций, в которых принимаются решения, и алгоритмы разработки рациональных решений.

В настоящее время под принятием решения (ПР) в ряде случаев понимают весь технологический цикл управления

ПР = С + А + П + ВР + Д + Р.

В ряде случаев под принятием решения понимается анализ, прогноз обстановки и выбор решения, т.е. отбрасываются этапы сбора информации и доведения информации, как рутинные операции, а оперативное управление (контроль за исполнением решения) рассматривается как самостоятельная задача

ПР = А + П + ВР.

И, наконец, под принятием решения часто понимают сам выбор решения, т.к. именно на этом этапе собственно и создается (синтезируется, генерируется) решение

ПР = ВР.

Решения можно различать в зависимости от длительности интервала времени $(t_o,t_f]$, на котором они реализуются. При этом различаются

- перспективные (долгосрочные, стратегические) решения, действующие на длительных временных интервалах, носящие глобальный характер и определяющие долговременные цели системы;
- оперативные решения - решения, которые действуют на небольших временных отрезках и предназначенные для реализации целевого назначения системы в конкретных условиях обстановки.

В зависимости от того, как часто необходимо принимать аналогичные решения в сходной обстановке, можно различать:

- однократные (уникальные) - решения, принимаемые единожды, обладающие существенной спецификой и действующие, как правило, на больших временных интервалах;
- повторяющиеся - решения (как правило, оперативные), которые принимаются периодически, причем изменение целей системы и обстановки ее функционирования незначительно.

1.2. Основные элементы и этапы принятия решения

Процесс принятия решений представляет собой достаточно сложный процесс, содержащий ряд этапов, при описании которых используются определенные элементы (понятия, конструкции). В качестве основных элементов принятия решений можно различать следующие:

1. Исследователь (субъект принятия решения) - один или несколько специалистов, проводящих анализ обстановки и выработку решения по управлению системой. Среди исследователей выделяют: лицо, принимающее решение (ЛПР) - специалист (командир), определяющий, что следует понимать под рациональным, желаемым в данной ситуации решением, выбирающий и утверждающий окончательное решение, а также несущий ответственность за данный выбор; лицо, формирующее решение (ЛФР) - специалист по способам формализации ситуации принятия решения и алгоритмам выбора рациональных решений. Довольно часто среди субъектов принятия решения не выделяют лицо, формирующее решение, и рассматривают только один элемент - лицо, принимающее решение.

2. С и с т е м а (объект принятия решения). В качестве системы может выступать реально существующая военно-техническая система, ее структура или какой-либо процесс, связанный с ее целевым функционированием.

3. Ц е л и. Целевой подход занимает в теории принятия решений центральное место. В задачах управления военно-техническими системами, проведения военной операции анализ обстановки начинается с выявления цели (целей) системы (операции). Принято различать две группы целей:

а). Цели проводимого системно-кибернетического исследования. Здесь в качестве такой цели рассматривается управление системой, проводимое на основе выбора (принятия) решений,

б). Цели системы. В задачах принятия перспективных решений по управлению (созданию) военно-технической системой должны быть выявлены основные цели, которые могут быть достигнуты при ее функционировании. Обычно целевой анализ начинается с уяснения (формулирования) так называемой глобальной цели. Глобальная цель конкретизируется путем указания подчиненных ей главных целей. В сложных задачах, решение которых зависит от многих взаимосвязанных факторов, целесообразным является дальнейшее развертывание главных целей в многоуровневое дерево целей и задач.

При принятии оперативных решений цель системы, как правило, очевидна и достаточно четко сформулирована.

4. Р е с у р с ы (о г р а н и ч е н и я). Функционирование военно-технической системы осуществляется в условиях различного рода ограничений, накладываемых обстановкой (средой), в которой должны быть реализованы принимаемые решения. Важнейшими видами ограничений являются ресурсные ограничения, в качестве которых выступают материальные, энергетические, информационные, людские, финансовые, временные и другие ограничения.

5. А л ь т е р н а т и в ы (альтернативные курсы действий) Различают следующие виды альтернатив:

- варианты поведения системы в различных условиях изменения состояния среды;
- варианты принятия решения в различных условиях обстановки.

Для оценки степени достижения целей, поставленных перед системой, необходимо уметь конструировать и анализировать альтернативные варианты поведения системы в различных условиях обстановки. В общем случае число таких вариантов бесконечно. Заметим, что в традиционных задачах исследования операций допустимое множество альтернатив обычно задается в явной или неявной форме. В современной теории принятия решений конструирование, генерация альтернатив и выявление множества допустимых альтернатив являются важной составной частью исследования.

В процессе принятия решения должны быть выявлены (сконструированы) допустимые с учетом введенных ограничений альтернативы (варианты решений) и выделена из них наилучшая с известной точки зрения.

В ситуациях, связанных с принятием перспективных решений, разработка каждой альтернативы поведения системы является достаточно трудоемкой операцией. Поэтому довольно часто разрабатываются наиболее вероятные варианты, число которых конечно, при этом каждой такой альтернативе обычно сопоставляется определенная последовательность действий, называемая курсом действий. При анализе альтернативных решений оцениваются последствия их выбора, а именно курсы действий.

6. Критерии (предпочтения, показатели). Критерий - это правило, по которому осуществляется выбор или сравнение альтернатив (решений). В качестве критерия выбора выдвигается либо условие принадлежности альтернативы к множеству, обладающему определенными свойствами, либо достижение при этой альтернативе экстремума некоторого показателя. При сравнении альтернатив в качестве критериев могут выступать отношения предпочтения, которые довольно часто задаются с использованием функций (целевых функций).

Основными задачами принятия решений, связанными с критериями, являются

- определение состава критериев (предпочтений, показателей);
- определение правил согласования критериев (нахождения компромисса между ними).

7. Модели. Исследование альтернатив и соответствующих им курсов действий производится на моделях. В современных системно-кибернетических исследованиях, как правило, возникает необходимость привлечения не одной модели, а нескольких разнотипных моделей, отражающих различные аспекты функционирования сложной системы, и проведения на этой осно-

ве многомодельных исследований. Типичные модели, используемые при решении различных задач:

- анализа - исследование качественных характеристик определенного варианта решения (альтернативы);
- синтеза - задача выбора эффективного (оптимального) решения (альтернативы) из множества допустимых решений.

Принятие решений осуществляется в несколько этапов. Это обусловлено тем, что в силу сложности решаемых проблем обычно не удается сразу учесть на каждом из этапов все необходимые факторы, и они могут быть выявлены только после окончания цикла исследований, что, в свою очередь, вызывает необходимость перехода к новому циклу (новой итерации). В действительности дело обстоит сложнее, т.к. существует необходимость внесения корректив в те или иные этапы по результатам выполнения других этапов, не дожидаясь окончания цикла исследования.

Принятие оперативных решений характерно тем, что в силу ограниченности ресурса времени, модели, необходимые для анализа и синтеза альтернатив разрабатываются заранее и уже учитывают во многом возможные изменения обстановки и возможные цели, достигаемые в различных условиях обстановки, складывающейся при функционировании ВТС.

Рассмотрим этапы принятия перспективных решений.

Первым этапом является уяснение цели функционирования системы в конкретных условиях обстановки. Необходимо четко обозначить качественные характеристики, которых необходимо добиться в результате управления. После этого целесообразно провести ретроспективный анализ, т.е., по возможности, проанализировать опыт решения подобных задач в прошлом. Следующим этапом является ресурсный анализ - анализ средств и ресурсов, необходимых для реализации решения, а также условий их использования. Т.к. решение вырабатывается на интервале $(t_р,t_д]$, а реализуется на интервале $(t_о,t_f]$ (причем, $t_д < t_о$), условия реализации решения могут измениться. Поэтому очень важно произвести прогноз изменения обстановки, в которой будет управляться система под воздействием принятого решения.

В результате выполнения первых четырех этапов разрабатывается, так называемая, концептуальная модель ситуации принятия решений, которая отражает с необходимой полнотой систему-прототип в том или ином содержательном ее аспекте и записана на естественном языке с использованием элементарных положений логики.

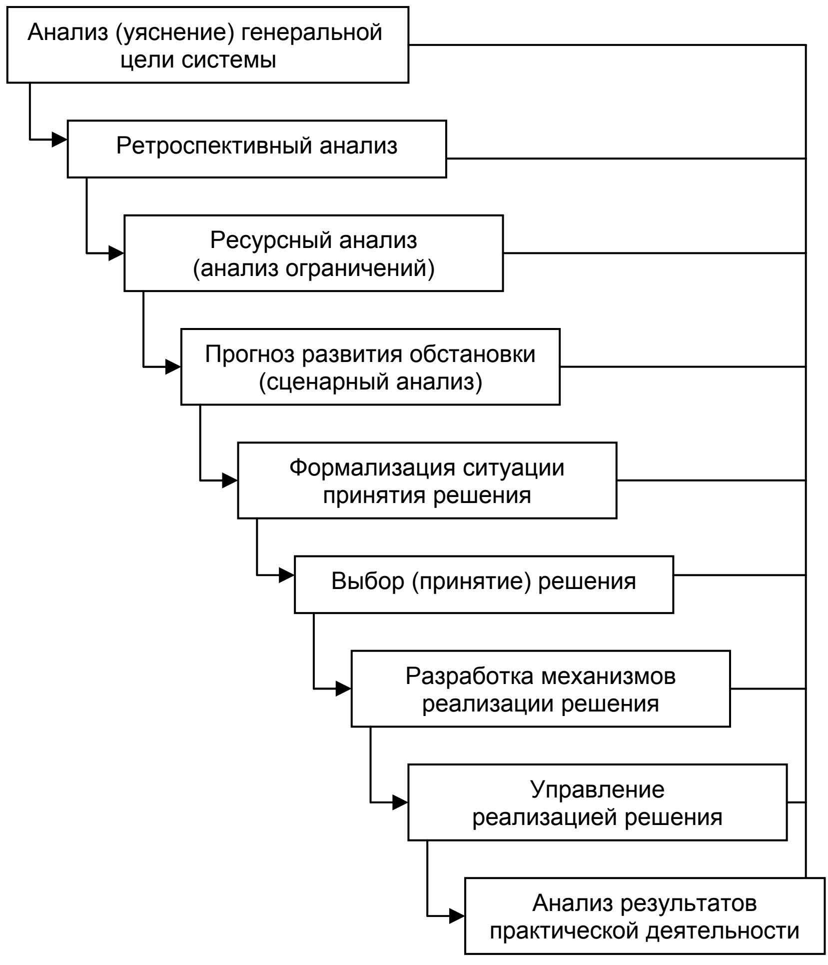


Рис. 1.2.

Теперь создается возможность формализованного описания ситуации, разработки модели. Модель описывает с необходимой для принятия решения полнотой особенности функционирования военно-технической системы, однако, для нахождения наилучшего в некотором смысле решения необходимо разрабатывать алгоритм - формальную последовательность действий, выполнение которой позволяет сформировать решение по управлению. При разработке алгоритмов широко используются методы оптимизации, развитые в математических дисциплинах. Для реализации решения (управления системой) необходимо разработать (или создать) механизмы управления системой. Качество разработанных моделей и алгоритмов принятия решений можно оценить на основе системотехнического анализа функционирования системы.

Анализируя содержание этапов принятия решения, можно заключить, что существует две группы проблем, определяющих весь облик системотехнического исследования систем.

Первая группа проблем (м о д е л ь н а я) связана разработкой модели - формализованного описания исходной ситуации принятия решения, для чего исследователь на основе глубокого системного анализа ситуации должен выбрать исходную математическую структуру принятия решения и в рамках этой структуры произвести формализованное описание условий, ограничивающих выбор, и предпочтений, которым должно удовлетворять оптимальное решение. Решение проблемы моделирования является в большой степени искусством и базируется, прежде всего, на опыте исследователя в решении подобных задач.

Вторая группа проблем (а л г о р и т м и ч е с к а я), непосредственно влияющая на первую группу, заключается в разработке алгоритмов выбора оптимального решения. Алгоритмы выбора на и м и т а ц и о н н ы х моделях используют процедуры случайного поиска, при этом моделируются различного рода внешние воздействия и варианты решений, из которых в последствии выбирается рациональное решение. Алгоритмы выбора решений, построенные на а н а л и т и ч е с к и х моделях, существенно различаются в зависимости от того, какой математической конструкцией описывается решение, и базируются на современных математических методах решения экстремальных задач различного рода.

Далее будем иметь в виду построение именно аналитических моделей принятия решений.

1.3. Проблемы моделирования при принятии решений

Принятие решений в военно-технических системах основывается на формализованном представлении ситуации - на математической модели. В деятельности человека моделирование как способ отражения объектов реальной действительности или мышления используется с глубокой древности. Однако широкое внедрение в практику управления военно-техническими системами абстрактных математических моделей началось в связи с успехами развития электронной вычислительной техники. В настоящее время математическое моделирование все чаще приобретает черты системного многомодельного исследования, а само понятие модели в процессе своего развития стало общенаучным, системно-кибернетическим понятием.

Существует точка зрения, что понятие модели относится к числу очень сложных понятий, поэтому вместо его определения следует просто показать, как осуществляется моделирование в различных областях научного знания. С этой точки зрения можно согласиться лишь применительно к начальному этапу изучения моделирования в тех или иных базисных дисциплинах. Что же касается таких дисциплин, как военно-техническая кибернетика, то для них всестороннее осмысление понятий "модель", "моделирование" имеет принципиальное значение, поскольку именно модель является основным инструментом исследования военно-технических систем в современных условиях. В самом широком смысле понятие модели можно дать на основе обобщения признаков, входящих в различные определения.

М о д е л ь - это система, исследование которой служит средством для получения информации о другой системе.

Здесь подчеркивается два основных свойства модели: первое - ее представление как системы (что является предпосылкой для дальнейшего развертывания системного подхода к моделированию); и, второе - ее главного назначения, как средства получения информации о некоторой системе (прототипе). Наряду с этим обобщенным определением целесообразно привести развернутую характеристику модели.

М о д е л ь - это некоторая промежуточная вспомогательная система (естественная или искусственная, материальная или абстрактная), обладающая следующими основными свойствами:

а) находится в объективном соответствии с познаваемым (изучаемым) объектом (системой - прототипом);

б) замещает в определенном отношении исследуемую систему;

в) дает информацию о системе, получаемую на основе исследования модели и соответствующих правил перехода модель - система (прототип).

Метод научного исследования систем, основанный на оперировании с моделями, называется моделированием. С другой стороны моделированием часто называют и процесс разработки модели.

Важной особенностью моделей принятия решений по сравнению с моделями, описывающими изменение состояния систем в процессе функционирования, является то, что они ориентированы на выбор решения, на использование некоторого алгоритма выбора. Это предъявляет дополнительные требования к формальным (математическим) способам описания ситуаций. Как правило, в силу неразвитости математических теорий выбора, модели принятия решений являются существенно более обобщенными по сравнению с моделями, описывающими функционирование систем.

Формально, с теоретико-множественных позиций модель M представляется отношением (совокупностью отношений), заданных на семействе m образующих множеств, т.е.

$M = (X_1, X_2,..., X_m, R)$,

здесь $R \subseteq X_1 \times X_2 \times \cdots \times X_m$ - график отношения.

Разработка модели принятия решения выполняется в несколько этапов.

Этапы разработки модели принятия решений.

1. Разработка концептуальной модели.

Модель, отражающая с необходимой полнотой прототип в том или ином содержательном аспекте и записанная на естественном языке с использованием положений логики, называется концептуальной моделью.

При разработке концептуальной модели довольно часто используются теоретико-множественные конструкции, которые позволяют в компактном виде представить основные особенности принятия решения. Часто используется аппарат теории графов.

2. Разделение характеристик, описывающих ситуацию принятия решения.

В ходе выполнения данного этапа выделяются:

- переменные, используемые при описании модели принятия решения; их можно разделить на неизвестные переменные, описывающие решение и интерпретируемые, как составляющие этого решения, и параметры - известные переменные, используемые при математическом описании модели;

- переменные, используемые при описании среды принятия решения.

Таким образом, производится разделение (декомпозиция) на собственно систему S и среду Q, тогда $R = S \circ Q$, где

$S = (X_1,.....,X_n, \Omega, S_g)$, где $S_g \subseteq X_1 \times X_2 \times \cdots \times X_n \times \Omega$,

$Q = (\Omega, X_{n+1},...,X_m, Q_g)$, где $Q_g \subseteq \Omega \times X_{n+1} \times X_{n+2} \times \cdots \times X_m$. Влияние среды Q на систему S описывается посредством переменных из Ω.

Прежде всего, здесь формализуется понятие "решение" в виде некоторой математической конструкции. Как правило, различают три основных формы представления решения:

1) решение задано в явном виде и представляет собой набор характеристик, непосредственно связанных со смыслом решаемой задачи;

2) решение описывается вектором - упорядоченным набором переменных, которые описывают суть решения в терминах исходной постановки задачи;

3) решение описывается функцией от времени (векторная функция), которая описывает изменение переменных задачи во времени.

Следует иметь в виду, что от удачного выбора переменных, описывающих решение, зависит простота модели, ее прозрачность и, следовательно, удобство ее дальнейшего анализа и использования. Далее в первом разделе (главы 2-7) будут рассмотрены алгоритмы нахождения оптимальных решений, описываемых некоторым вектором, заданным в конечномерном пространстве; во втором разделе (главы 8-11) будут рассмотрены алгоритмы нахождения оптимальных решений, описываемых функцией от времени.

3. Анализ информации, имеющейся о параметрах, описывающих среду принятия решения.

В зависимости от степени знания среды принятия решения будем различать следующие модели среды:

- детерминированная - имеется полная информация о параметрах среды и их влиянии на процесс выбора решения;

- недетерминированная среда характерна тем, что конкретное состояние среды не известно. Здесь различают следующие виды среды:

- стохастическая - в массовых проявлениях воздействия среды имеется устойчивость появления тех или иных ее состояний (свойство статистической устойчивости).
- целенаправленная - не известно конкретное воздействие на систему, но известна цель этого воздействия;
- неизвестная.

Результаты выполнения данного этапа определяют вид модели и во многом влияют на последующий выбор алгоритма выбора решения.

4. Разработка формализованной модели, описывающей допустимые варианты принимаемых решений - множество допустимых решений.

Формализация требований, предъявляемых к принимаемому решению, в виде ограничений, которым оно должно удовлетворять. В совокупности такие ограничения задают множество допустимых альтернатив, т.е. множество из которого выбирается решение. При этом нужно следить, чтобы в модель были включены все ограничительные условия, и в то же время не было ни одного лишнего или записанного в более жесткой, чем это требуется условиями задачи, форме.

В задачах математического программирования решение описывается некоторым вектором, а множество допустимых решений задается ограничениями, представляющими собой функциональные неравенства (равенства). Соответствующие модели принятия решения принято называть с т а т и ч е с к и м и.

В задачах оптимального управления решение описывается функциями времени (состояния, выхода). Соответствующие модели принятия решения называют д и н а м и ч е с к и м и.

5. Формализация правила выбора решения в форме ограничений или целевых функций (функционалов).

В зависимости от формы, в которой предъявляются требования к принимаемому решению, различают выбор

- сатисфакционный (решение удовлетворяет некоторым ограничениям);
- оптимизационный (выбирается наилучшее решение).

В результате выполнения рассмотренных этапов разрабатывается математическая модель принятия решений. Анализируя особенности принятия решений при управлении военно-техническими системами, можно заключить, что в этом процессе взаимодействуют три основных элемента:

- исследователь - лицо, принимающее решение (ЛПР);
- система (процесс), относительно которой принимается решение;
- среда - окружающая действительность, в которой принимается решение.

Тогда факторы, определяющие проблемы моделирования при принятии решения, связаны именно с этими элементами (см. раздел 3). В частности, это

- разнородные требования, предъявляемые ЛПР к процессу функционирования системы, приводят к множественности оценок качества процесса управления, что требует использования нескольких показателей качества решения (нескольких критериев выбора решения);
- система, относительно которой принимается решение, может быть достаточно сложна, иметь развитую иерархическую структуру, отражающую различные отношения подчиненности органов управления, что вносит определенную сложность в процесс принятия решений в таких системах и требует разработки специфических моделей и алгоритмов принятия решений;
- влияние среды, воздействующей на процесс целевого функционирования системы и процесс управления системой, как правило, не известно достаточно точно, что вносит дополнительные сложности в процесс принятия решения.

С проблемами адекватного моделирования систем неизбежно приходится сталкиваться при решении задач управления. В современных системно-кибернетических исследованиях выделяют два основных принципа, на основе которых решаются указанные проблемы. Это принципы декомпозиции и агрегирования.

Д е к о м п о з и ц и я - расчленение системы на части (подсистемы), при котором исследование системы может быть проведено на основе исследования подсистем, при этом адекватность обеспечивается за счет введения в подсистемы сигналов или нагрузок, учитывающих взаимодействие выделяемых при декомпозиции подсистем. В частности, на основе данного принципа осуществляется выделение собственно системы, относительно которой осуществляется принятие решения, и среды, оказывающей влияние на процессы принятия решений в системе.

А г р е г и р о в а н и е - обобщение системы в том или ином смысле, связанном с целями исследования. Агрегированная система соответствует исходной системе в главных, интересующих исследователя чертах. Агрегирование является концептуальной основой моделирования, т.к. модель всегда представляет собой агрегативный (обобщенный) образ системы, отражающий ее в главных, интересующих исследователя чертах.

1.4. Обобщенная математическая модель принятия решения (математическая структура выбора)

С целью рассмотрения на единой концептуальной основе многочисленных задач выбора, проведения соответствующих классификаций, установления связей между задачами различных классов целесообразно использовать при общей постановке задач принятия решений структурно-математический подход [40]. При построении конкретной модели необходимо, прежде всего, иметь ввиду требование достижения соответствия математического описания проблемы принятия решений той обстановке, в которой это решение реализуется.

Математическая структура выбора (постановка задачи принятия решений) в общем виде может быть представлена как:

$$(Q(s), \Delta, \{ r_i, i \in C \}, \{ f_j, j \in G \}) . \qquad (1.1)$$

Здесь

$Q(s)$ - исходная структура выбора (модель); s - тип структуры. Структура $Q(s)$ позволяет ставить задачи выбора, связанные с теми или иными структурными (модельными) ограничениями, задаваемыми посредством сетей, алгебраических уравнений (статические модели), посредством дифференциальных уравнений для динамических систем и т.д.

Δ - пространство альтернатив (решений). В зависимости от типа исходной структуры выбора это некоторое конечномерное пространство векторов или пространство векторных функций, отдельный элемент которого характеризует структуру решения (альтернативы), состав и смысл компонент решения определяется существом решаемой задачи.

$\{r_i, i \in C\}$ - множество отношений, ограничивающих выбор; C - множество индексов отношений, ограничивающих выбор. Данные отношения задаются на Δ и вводятся непосредственно в процессе постановки задач выбора, они отражают основные пространственно-временные, технические и технологические ограничения, связанные с функционированием военно-технических систем. Ограничивающие отношения часто представляют в виде равенств или неравенств, что позволяет связать постановку отдельных задач выбора с задачами математического программирования или оптимального управления.

$\{f_j,\ j\in G\}$ - множество отношений предпочтения, задаваемых на Δ и отражающих различные требования, предъявляемые к наилучшему решению; G - множество индексов отношений предпочтения. В зависимости от конкретного вида $f_j,\ j\in G$ различают задачи кардинального, псевдокардинального или ординального выбора. Так, если $f_j,\ j\in G$ являются функциями $f_j : \Delta \to R$, то такие задачи называют задачами к а р д и н а л ь н о г о выбора. Если же $f_j,\ j\in G$ являются функциями $f_j : \Delta \to N$, где N - множество натуральных чисел (т.е. каждой альтернативе сопоставляется определенное число баллов), то говорят о задачах п с е в д о к а р д и н а л ь н о г о выбора. В случае, когда $f_j,\ j\in G$ представляют собой отношения порядка $f_j \subseteq \Delta\times\Delta$, говорят об о р д и н а л ь н о м (групповом) выборе.

Данная постановка (математическая структура) представляет широкие возможности для того, чтобы в ее рамках рассматривать разнообразные задачи принятия решений. Рассмотренные здесь элементы всегда явно или неявно присутствуют при выборе наилучшего решения. Вместе с тем в ряде случаев целесообразно упростить рассмотренную постановку, оставив в ней наиболее существенные конструкции. Так, множество альтернатив Δ можно непосредственно связать с математической моделью выбора и обозначать его как Δ_s, а, выделяя из этого множества подмножество альтернатив, удовлетворяющих отношениям, ограничивающим выбор, можно сформировать множество д о п у с т и -м ы х альтернатив $\Delta_{sr} \subseteq \Delta_s$. Именно из Δ_{sr} необходимо выбирать наилучшее решение в соответствии с заданными отношениями предпочтения, причем наиболее простая ситуация принятия решения складывается, когда таких отношений всего одно, т.е. $|G| = 1$. В этом случае рассмотренная математическая структура выбора имеет наиболее простой вид (Δ_{sr}, f).

Далее, как правило, Δ_{sr} будем обозначать для простоты, как Δ. Тогда задачу выбора оптимального решения x^* можно записать

$$x^* = \arg \underset{x\in\Delta}{\operatorname{opt}} f(x). \qquad (1.2)$$

Здесь opt f(x) - к р и т е р и й (правило) выбора оптимального решения, в конкретных задачах он имеет вид max f(x) или min f(x);

arg - имеет смысл выделения какого-либо аргумента функции opt f(x) (в отличие от Arg, что соответствует выделению всего множества оптимальных решений);

f(x) - ц е л е в а я (критериальная) функция, причем конкретное значение этой функции $a = f(x)$ часто называют п о к а з а т е л е м качества решения x;

Δ - множество д о п у с т и м ы х альтернатив (допустимых в смысле исходной модели (структуры) выбора и отношений, ограничивающих выбор).

Тогда задачу принятия решения (1.2) кратко можно характеризовать парой

$$(\Delta, f)\ , \tag{1.3}$$

здесь Δ - множество допустимых альтернатив;

f - правило (критерий) выбора решения.

В зависимости от того, каким образом задаются элементы Δ и f, различают различные математические модели и задачи принятия решений. В частности, если решение описывается некоторым вектором, то соответствующие модели называют с т а т и ч е с к и м и, а алгоритмы выбора наилучшего решения используют методы математического программирования, если же решение описывается функцией времени, то модели называют д и н а м и ч е с к и м и, а алгоритмы строятся на основе методов оптимального управления.

Контрольные вопросы

1). В чем состоит необходимость проведения системно-кибернетических исследований ?

2). Какое место занимают вопросы принятия решений при управлении военно-техническими системами ?

3). Какие проблемы стоят при принятии решений, и с использованием каких принципов они решаются ?

4). Какие основные элементы должна содержать математическая постановка задачи принятия решений. В чем их роль ?

5). Какими математическими конструкциями описывается решение в статических и динамических моделях ?

6). Какие методы оптимизации могут применяться при использовании статических моделей принятия решений ?

7). Какие методы оптимизации применяются при использовании динамических моделей принятия решений ?

РАЗДЕЛ 1. ПРИНЯТИЕ РЕШЕНИЙ НА ОСНОВЕ СТАТИЧЕСКИХ МОДЕЛЕЙ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМ

2. ПОСТАНОВКА ЗАДАЧ ВЫБОРА РЕШЕНИЙ В КОНЕЧНОМЕРНЫХ ВЕКТОРНЫХ ПРОСТРАНСТВАХ

2.1. Особенности задач математического программирования

Оптимизация процессов управления военно-техническими системами на основе аналитических статических моделей принятия решений связана с поиском некоторого конечномерного вектора, описывающего решение, который доставляет оптимум (максимум или минимум) целевой функции. Алгоритмы нахождения таких векторов, основанные на использовании методов оптимизации, развиваются в рамках математического программирования. Содержание математического программирования составляют теория и методы решения задач нахождения экстремумов функций на множествах, определяемых в n-мерном евклидовом пространстве и заданных линейными и нелинейными ограничениями (равенствами и неравенствами). Здесь, прежде всего, различают задачи с непрерывными переменными (компоненты вектора, описывающего решение, действительные числа) и задачи с целочисленными переменными. В зависимости от характера целевой функции и ограничений различают задачи линейные и нелинейные. В целом с учетом специфики используемых методов оптимизации в математическом программировании принято выделять:

а) л и н е й н о е программирование - целевая функция линейна, а множество, на котором ищется ее экстремум, задается системой линейных равенств и неравенств;

б) н е л и н е й н о е программирование - нелинейны целевая функция или ограничения.

в) д и с к р е т н о е программирование - осуществляется выбор оптимального решения в ситуациях, когда на компоненты вектора, описывающего решение, накладываются ограничения целочисленности (в общем случае выбор производится на дискретных (целочисленных) множествах); такие задачи также можно разделить на линейные и нелинейные.

В нелинейном программировании принято выделять в ы п у к л о е программирование - целевая функция вогнута (выпукла вверх) (решается задача ее максимизации) и выпукло множество, на котором решается экстремальная задача. В дискретном программировании выделяют ц е л о ч и с л е н н о е программирование - экстремальные задачи, в которых компоненты вектора, описывающего решение, должны быть целочисленны. Частным случаем целочисленного программирования является б у л е в о е (б и в а л е н т н о е, л о г и ч е с к о е) программирование - компоненты вектора, описывающего решение, должны принимать значения 0 или 1.

В задачах математического программирования можно конкретизировать составляющие обобщенной модели принятия решения (выбора) (1.1)

$(Q(s), \Delta, \{ r_i, i \in C \}, \{ f_j, j \in G \})$.

$Q(s)$ - исходная структура выбора, задается функциональными ограничениями.

Δ - пространство альтернатив, представляет собой пространство векторов размерности (n); $\Delta = R^n$; если $x \in \Delta$, то $x = (x_1, x_2,..., x_n)^T$.

$\{r_i, i \in C\}$ - отношения, ограничивающие выбор, задаются некоторыми функциями: $\varphi_i(x) \le b_i$, $i \in C = \{1,\ldots, m\}$.

$\{f_j, j \in G\}$ - отношения предпочтения альтернатив - правило выбора (в задачах математического программирования $|G|=1$) задается функцией (целевой функцией) $f: \Delta \to R$; решение x_s более предпочтительно, чем решение x_q, если значение целевой функции $f(x_s) > f(x_q)$.

Тогда обобщенная модель выбора (принятия оптимального решения) x^* в форме задач математического программирования будет иметь вид

$$x^* = \arg \max_{x \in \Delta} f(x), \text{ где } \Delta = \{x \in R^n \mid \varphi_i(x) \le b_i,\ i = 1,\ldots, m \}. \quad (2.1)$$

Здесь Δ - множество допустимых решений (альтернатив), т.е. тех решений из $\Delta \subseteq R^n$, которые удовлетворяют отношениям, ограничивающим выбор (отражают специфические особенности исходной задачи и принципиально могут быть реализованы). Из множества Δ необходимо выбрать наиболее предпочтительное решение x^*, которое доставляет максимальное значение целевой функции f(x).

Так как содержанием задач математического программирования является теория и методы нахождения экстремумов функций на множествах, определяемых в n-мерном евклидовом пространстве, напомним ряд математических понятий и определений, которые будут использоваться в дальнейшем.

♦ R^n называется л и н е й н ы м (векторным) пространством, если 1). $\forall x,y \in R^n \quad x + y \in R^n$

2). $\forall \alpha \in R \quad \alpha x \in R^n$

♦ R^n называется е в к л и д о в ы м пространством (часто его обозначают, как E^n), если $\forall x,y \in R^n$ определено скалярное произведение

$$(x,y) = x^T y = \sum_{i=1}^{n} x_i y_i$$

♦ Е в к л и д о в о й н о р м о й (длиной вектора) x называется

$$\|x\| = (x.x)^{1/2}$$

♦ Расстоянием d(x,y) между векторами x и y называется евклидова норма разности x и y

$$d(x,y) = \|x - y\|$$

♦ Множество $U(x,\varepsilon)$ называется ε - о к р е с т н о с т ь ю (окрестностью) точки x, если

$$U(x,\varepsilon) = \{y \mid d(x,y) = \|x - y\| \le \varepsilon \}$$

♦ Точка (решение) x* называется точкой л о к а л ь н о г о максимума, если

$$\forall \varepsilon > 0 \quad \forall (x \in \Delta \cap U(x,\varepsilon)) \ (f(x) \le f(x^*))$$

♦ Точка (решение) x* называется точкой г л о б а л ь н о г о максимума, если

$$x^* = \arg \max_{x \in \Delta} f(x).$$

♦ Задача выбора на Δ называется о д н о э к с т р е м а л ь н о й, если локальный максимум достигается в единственной точке (она же является и точкой глобального максимума). В противном случае задача выбора называется м н о г о э кс т р е м а л ь н о й.

♦ Задача выбора называется задачей выбора б е з у с л о в н о г о экстремума, если $\Delta = R^n$.

♦ Задача выбора называется задачей выбора у с л о в н о г о экстремума, если $\Delta \subset R^n$.

♦ Последовательность альтернатив $\{x^k, k=1,2,\ldots\}$, $\forall x^k \in \Delta$ называется м а к с и м и з и р у ю щ е й, если $\forall i<j \quad f(x^i) < f(x^j)$.

♦ Последовательность альтернатив $\{y^k, k=1,2,\ldots\}$, $\forall y^k \in \Delta$ называется м и н и м и з и р у ю щ е й, если $\forall i<j \quad f(y^i) > f(y^j)$.

♦ С у п р е м у м о м $f(x)$ на Δ называется предел значений $f(x)$ на максимизирующей последовательности альтернатив

$$\sup_{x\in\Delta} f(x) = \lim_{k\to\infty} f(x^k).$$

♦ И н ф и н у м о м $f(y)$ на Δ называется предел значений $f(y)$ на минимизирующей последовательности альтернатив

$$\inf_{x\in\Delta} f(y) = \lim_{k\to\infty} f(y^k).$$

Если $\lim\limits_{k\to\infty} f(x^k) = f(x^s)$ и $x^s \in \Delta$, то $\sup\limits_{x\in\Delta} f(x) = \max\limits_{x\in\Delta} f(x)$.

Если $\lim\limits_{k\to\infty} f(y^k) = f(y^s)$ и $y^s \in \Delta$, то $\inf\limits_{x\in\Delta} f(x) = \min\limits_{x\in\Delta} f(x)$.

2.2. Примеры моделей принятия решения в виде задач математического программирования

Рассмотрим некоторые достаточно часто используемые формализованные модели принятия решений.

<u>1. Общая сетевая модель оптимизации</u>

Пусть задана некоторая сеть N, описывающая возможный обмен в системе (возможные потоки грузов, информации и т.д.), в виде N = (A, E, f), где

$A = \{A_1, A_2, \ldots , A_n\}$ - множество вершин (узлов) сети;

$A \subseteq A \times A$ - множество дуг сети;

$f: E \to R^1$ - вещественная скалярная функция, ставящая в соответствие каждой дуге $<A_i, A_j>$ некоторый показатель качества f_{ij} (время, стоимость, затраты энергии, и т.д.).

Для характеристики сетей в прикладных оптимизационных задачах необходимо ввести еще два рода величин. Это, во-первых, чистый поток T_k узла k, характеризующий, как изменится поток при прохождении узла k, и, во-вторых, это показатель пропускной способности u_{ij} дуги $<A_i, A_j>$.

Тогда набор переменных x_{ij}, характеризующих величину потока, проходящего по дуге $<A_i, A_j>$, представляет собой некоторый вариант организации обмена в сети. Естественно, что поток x_{ij} не должен превышать пропускной способности дуги u_{ij}. Кроме того, разность входящего в k-ый узел потока и потока, выходящего из него, должна равняться T_k. В этих условиях целесообразно искать поток, минимизирующий некоторое суммарное качество (время, стоимость, и т.п.).

Весьма общей оптимизационной сетевой задачей, имеющей многочисленные практические приложения, является задача следующего вида

$$\sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij}\, x_{ij} \to \min$$

при ограничениях

$$\sum_{j \in J_k^+}^{n} x_{jk} - \sum_{j \in J_k^-}^{n} x_{kj} = T_k,\ k = 1,\dots,n.$$

$$0 \le x_{ij} \le u_{ij},\quad \forall \langle A_i, A_j\rangle \in E$$

Здесь J^+_k - множество вершин, смежных с к-ой, из которых идут дуги в к-ю вершину; J^-_k - множество вершин, в которые идут дуги из к-ой вершины;

u_{ij} – пропускная способность дуги $\langle A_i, A_j\rangle$.

2. Классическая транспортная модель

Транспортная модель является частным случаем общей сетевой модели оптимизации и описывает возможные варианты перемещения однородного ресурса (люди, машины, самолеты, информация и т.д.) между поставщиками и потребителями. Тогда A множество узлов сети можно разбить на два подмножества P и H, причем $A = P \cup H$; $P \cap H = \varnothing$; здесь

P - множество поставщиков; $P = \{p_i,\ i = 1,\dots, p\}$;

H - множество потребителей; $H = \{h_j,\ j = 1,\dots, h\}$;

Множество дуг классической транспортной модели задается следующим образом: $E = P \times H$, т.е. каждый поставщик соединен с каждым потребителем. Задана функция $f: E \to R^1$, характеризующая некоторое качество перевозок (время, стоимость, расход топлива,…). Каждому поставщику соответствует объем имеющегося у него ресурса P_i, а каждому потребителю сопоставляется объем необходимого для него ресурса H_j.

В этом случае в качестве математической конструкции, описывающей решение, целесообразно выбрать вектор $x = \| x_{ij} \|$, компоненты которого характеризуют объем перевозок от поставщика p_i к потребителю h_j. Естественно, что от i-го поставщика невозможно вывести продукции больше P_i (больше, чем у него имеется). С другой стороны необходимо удовлетворить запросы j-го потребителя в H_j. В этих условиях следует находить решение, минимизирующее суммарные расходы на перевозки.

Тогда математическая модель классической транспортной задачи имеет вид:

$$\sum_{i=1}^{p} \sum_{j=1}^{h} f_{ij}\, x_{ij} \to \min$$

при ограничениях

$$\sum_{j=1}^{h} x_{ij} \le P_i,\ i = 1,\ldots, p,$$

$$\sum_{i=1}^{p} x_{ij} \ge H_j,\ j = 1,\ldots, h,$$

$$x_{ij} \ge 0,\quad i = 1,\ldots, p;\ j = 1,\ldots,h.$$

Отметим, что если имеется равенство суммарного ресурса у всех поставщиков и суммарных потребностей всех потребителей,

$$\sum_{i=1}^{s} P_i = \sum_{j=1}^{q} H_j\ ,$$

то неравенства в ограничениях транспортной модели приобретают вид равенств.

3. Модель назначений

Задачу о назначениях можно кратко сформулировать следующим образом. Задано n работ, каждую из которых может выполнить любой из n исполнителей, качество выполнения i-й работы j-м исполнителем характеризуется f_{ij}. Необходимо назначить исполнителей для выполнения работ таким образом, чтобы суммарное качество выполнения всех работ было максимальным. Модель назначений является частным случаем классической транспортной модели, в которой (m = n) и $\forall P_i = 1$, $\forall H_i = 1$.

Здесь решение целесообразно описывать вектором x, компоненты которого x_{ij} принимают значения 0 или 1 в зависимости от того назначен j-ый исполнитель на выполнение i-ой работы или нет. Тогда в качестве ограничений выступают естественные условия невозможности назначения одного исполнителя для выполнения различных работ, и условия невозможности выполнения работы несколькими исполнителями. Целесообразно находить распределение исполнителей, максимизирующее суммарное качество выполнения всех работ.

Математическая модель назначений имеет вид:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij}\, x_{ij} \to \max$$

при ограничениях

$$\sum_{j=1}^{n} x_{ij} = 1,\ i = 1,\ldots, n,$$

$$\sum_{i=1}^{n} x_{ij} = 1,\ j = 1,\ldots, n,$$

$$x_{ij} \in \{0,1\}\ ,\ i = 1,\ldots, n;\ j = 1,\ldots, n.$$

К такой же модели могут быть сведены ситуации, когда исполнителей больше, чем работ, или наоборот.

4. Модель оптимального планирования выпуска продукции

Пусть предприятие выпускает (n) наименований продукции, в производстве которых используется (m) видов ресурсов.
Обозначим:

a_{ij} – объем затрат i-го вида ресурса на производство единицы j-го вида продукции; i = 1,..., m; j = 1,..., n;

b_i - полные объемы ресурсов, имеющихся на предприятии; i = 1,...,m;

c_j - стоимость реализации j-го вида продукции на рынке товаров; j = 1,.. ..., n;

d_j - целесообразный объем выпуска продукции j-го вида; j = 1,.. ..., n.

Необходимо составить такой план выпуска продукции, который удовлетворяет ресурсным ограничениям и минимизирует отклонения от целесообразного объема выпуска.

В этом случае план выпуска продукции можно описать вектором $x = \| x_j \|$, где x_j характеризует объем выпуска j-го вида продукции. В качестве ограничений выступает требование выпуска продукции в условиях имеющихся ресурсов. Здесь целесообразно находить решение, минимизирующее суммарные отклонения объемов выпуска продукции от заданных.

Математическая модель такой задачи имеет вид:

$$\sum_{j=1}^{n} (x_j - d_j)^2 \to \min$$

$$\sum_{j=1}^{n} a_{ij}\, x_j \le b_i\ , \quad i=1,...,m;$$

$$x_j \ge 0,\ \ j = 1,\ldots,n.$$

В случае, когда оптимальное значение целевой функции такой задачи равно 0 (ресурсов достаточно для выпуска всех видов продукции), задачу планирования оптимального выпуска продукции целесообразно ставить в виде:

$$\sum_{j=1}^{n} c_j\, x_j \to \max$$

$$\sum_{j=1}^{n} a_{ij}\, x_j \le b_i\ , \quad i=1,...,m;$$

$$d_j \le x_j \le D_j,\ \ j = 1,\ldots,n.$$

Здесь D_j – верхний предел целесообразного выпуска продукции j-го вида.

5. Модель оптимального программного управления линейным динамическим объектом

Задача нахождения оптимальной программы управления линейным динамическим объектом может быть решена методами оптимального управления динамическими системами (методы, основанные на принципе максимума Л.С.Понтрягина или на принципе оптимальности Р.Беллмана). Однако в ряде случаев исходную динамическую модель целесообразно свести к статистической и использовать для ее решения методы математического программирования, поскольку в этом случае многие сложные ограничения, накладываемые как на переменные состояния, так и на допустимые управления, которые крайне усложняют двухточечную краевую задачу, легко учитываются алгоритмами математического программирования.

Пусть изменение состояния линейного объекта управления описывается системой дифференциальных уравнений

$$\dot{x}(t) = A(t)x + B(t)u\ ,$$

где x - n-мерный вектор состояния;

u - r-мерный вектор управления, под воздействием которого изменяется состояние.

Необходимо найти программу управления u=u(t) на интервале времени $t\in(t_o,t_f]$, позволяющую перевести линейный динамический объект из состояния $x(t_o)$ в состояние $x(t_f)$, и доставляющую максимум показателю качества J(x,u)

$$J(x,u) = \int_{t_0}^{t_f} F(t,x,u)\,dt \to \max$$

Возможные состояния и управления должны удовлетворять ограничениям $x(t)\in X$, $u(t)\in U$, $t\in(t_o,t_f]$,, где X и U - соответственно множества допустимых состояний и управлений.

Состояние линейного объекта управления в любой момент времени определяется в соответствии с формулой Коши как

$$x(t) = \Phi(t,t_o)\, x(t_o) + \Phi(t,t_o) \int_{t_0}^{t} \Phi^{-1}(\tau,t_o)\, B(\tau)\, u(\tau)\, d\tau\ ,$$

где $\Phi(t,t_o)$ - $n\times n$-мерная фундаментальная матрица решений однородной системы дифференциальных уравнений (переходная матрица).

Проводя дискретизацию по времени, т.е. разбивая весь интервал $(t_o,t_f]$ на N подынтервалов, последнее выражение можно переписать

$$x(k) = \Phi(t_k,t_{k-1})x(k-1) + \Phi(t_k,t_{k-1}) \int_{t_{k-1}}^{t_k} \Phi^{-1}(\tau,t_{k-1})B(\tau)d\tau\; u(k-1),\ k = 1,...,N$$

Тогда задача нахождения оптимального программного управления будет иметь вид:

$$\sum_{k=1}^{N} F(x(k),u(k-1)) \to \max$$

$$x(k) = \Phi(t_k,t_{k-1})x(k-1) + \Phi(t_k,t_{k-1}) \int_{t_{k-1}}^{t_k} \Phi^{-1}(\tau,t_{k-1})B(\tau)d\tau\; u(k-1),\ k = 1,...,N$$

$$x(k) \in X,\ u(k-1) \in U,\ k = 1,\dots,N,$$

$$x(0) = x(t_o),\ x(N) = x(t_f)\ .$$

Данная задача имеет $N_o = N\times(m+r)$ переменных, соответствующих различным значениям $x(k)$, $u(k-1)$, $k = 1,...,N$. Вводя новый вектор неизвестных переменных y, размерности N_o задачу оптимального программного управления линейным динамическим объектом можно представить в форме задачи математического программирования. Размерность такой задачи при большом N может быть достаточно велика, тем не менее, в ней достаточно просто учитываются сложные ограничения на возможные состояния и управления линейного динамического объекта.

Рассмотренные примеры задач математического программирования относятся к задачам различных классов. Задачи 1, 2 являются задачами линейного программирования, задача 3 является задачей дискретного (булевого) программирования, задачи 4, 5 представляют собой задачи нелинейного программирования.

2.3. Элементы выпуклого анализа задач оптимизации

Итак, рассматриваем задачу выбора решения в конечномерных пространствах, поставленную в виде (2.1)

$$x^* = \arg \max_{x\in\Delta} f(x), \text{ где } \Delta = \{x\in R^n \,|\, \varphi_i(x)\leq b_i,\ i = 1,\ldots, m\}.$$

Рассмотрим ряд определений и утверждений, связанных с заданием свойств Δ и $f(x)$, которые используются в выпуклом анализе для исследования таких задач оптимизации.

Пусть задано множество $D \subset R^n$.

♦ Точка $x\in R^n$ называется *предельной* точкой множества D, если в любой окрестности этой точки содержится бесконечно много точек из D, отличных от x.

♦ Если точка $x\in D$ не является предельной, то она называется *изолированной*.

♦ Точка $x\in D$ называется *внутренней*, если существует такая ее ε-окрестность $U(x,\varepsilon)$, что все точки этой ε-окрестности принадлежат D; $U(x,\varepsilon) \subset D$.

♦ Точка $x\in D$ называется *граничной*, если в любой ε- окрестности этой точки содержатся как точки из D, так и точки, не принадлежащие D. Множество, состоящее из всех граничных точек называется *границей* D.

♦ Множество $D \subset R^n$ называется з а м к н у т ы м, если оно содержит все свои предельные точки.

♦ Множество $D \subset R^n$ называется о г р а н и ч е н н ы м, если $\exists C>0 \mid \forall x \in D \ \|x\| < C$.

♦ Множество $D \subset R^n$ называется в ы п у к л ы м, если для $\forall x,y \in D$ оно содержит отрезок, соединяющий эти точки, т.е.

$$\forall x,y \in D \ \ \forall \alpha \in [0,1] \ \ (\alpha x + (1-\alpha)y) \in D$$

♦ Пересечение выпуклых множеств является выпуклым множеством.

♦ Ограниченное замкнутое множество называется к о м п а к т н ы м.

Отделимость выпуклых множеств

♦ Г и п е р п л о с к о с т ь ю Г в R^n называется множество вида

$$Г = \{ x \in R^n \mid (c,x) = a \}, \quad \text{где } c \neq 0 .$$

В пространстве R^n гиперплоскость Г задает:

- два замкнутых полупространства: $Г^{(-)} = \{ x \in R^n \mid (c,x) \leq a\}$, $Г^{(+)} = \{ x \in R^n \mid (c,x) \geq a\}$
- два открытых полупространства: $G^{(-)} = \{ x \in R^n \mid (c,x) < a\}$, $G^{(+)} = \{ x \in R^n \mid (c,x) > a\}$

♦ О п о р н о й гиперплоскостью Г выпуклого множества D в некоторой его граничной точке (x) называется гиперплоскость, проходящая через (x) и разбивающая R^n на два полупространства $Г^{(+)}$ и $Г^{(-)}$, в одном из которых полностью содержится D.

♦ Множества D и Q называются с т р о г о о т д е л и м ы м и друг от друга, если существует такая гиперплоскость Г, что множества D и Q принадлежат различным открытым полупространствам, определяемым Г.

♦ Множества D и Q называются с о б с т в е н н о о т д е л и м ы м и, если они принадлежат различным закрытым полупространствам.

♦ Любые два непересекающиеся, выпуклые множества строго отделимы.

Таким образом, если D представляет собой объединение непересекающихся, выпуклых множеств, то поиск оптимального решения можно искать на каждом множестве независимо от других. Глобальный оптимум находится путем сравнения оптимальных выборов на каждом множестве.

Представление выпуклых множеств

♦ Точка x называется л и н е й н о й к о м б и н а ц и е й точек $x_1, x_2,..., x_k$, если существуют такие числа $\alpha_1, \alpha_2,\ldots, \alpha_k \in R^1$, что

$$x = \sum_{i=1}^{k} \alpha_i x_i$$

Если же $\forall \alpha_l \geq 0$ и $\sum_{i=1}^{k} \alpha_i = 1$, то линейная комбинация точек называется в ы п у к л о й комбинацией.

♦ Точка (x) множества D называется к р а й н е й (угловой) точкой, если не существует точек $y,z \in D$, таких, что

$$x = (\alpha y + (1-\alpha)z) \quad \text{при некотором } \alpha\in(0,1).$$

Т.е. не существует в D отрезка с концами y,z, для которого (x) была бы внутренней точкой.

Утверждение.

Любая точка (x) выпуклого, замкнутого и ограниченного множества D может выть представлена в виде выпуклой комбинации конечного числа крайних точек этого множества.

♦ Количество крайних точек, необходимых для представления точки (x) выпуклого, замкнутого и ограниченного множества D, не превышает числа $p = \dim(D) + 1$.

Таким образом, множество всех допустимых решений Δ в задаче математического программирования можно представить в виде выпуклой комбинации крайних точек Δ.

Выпуклые функции. Дифференцируемость

♦ Функция $f: D \to R^1$, заданная на выпуклом множестве $D \subseteq R^n$, называется в ы п у к л о й (выпуклой вниз) на D, если

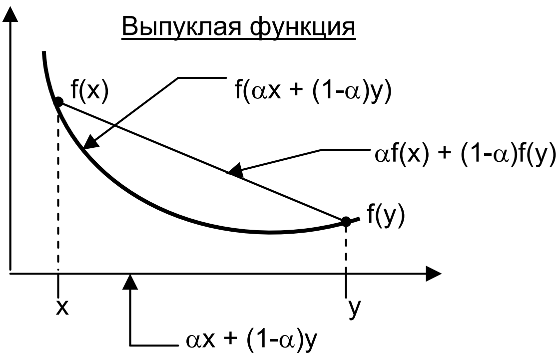
$$\forall x,y \in D, \forall \alpha\in(0,1) \quad f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y),$$

и называется с т р о г о в ы п у к л о й , если

$$\forall x,y \in D, \forall \alpha\in(0,1) \quad f(\alpha x + (1-\alpha)y) < \alpha f(x) + (1-\alpha)f(y).$$

♦ Если f(x) - выпуклая функция, то h(x) = -f(x) - вогнутая (выпуклая вверх) функция. Действительно, тогда

$$\forall x,y \in D, \forall \alpha\in(0,1) \quad h(\alpha x + (1-\alpha)y) \geq \alpha h(x) + (1-\alpha)h(y).$$



♦ Л и н и я у р о в н я (с) функции f(x) - множество точек вида $\{x \in D \mid f(x) = c\}$.

♦ Множество $A = \{x \in D \mid f(x) \le c\}$, ограниченное линией уровня выпуклой функции f(x), заданной на выпуклом множестве D, выпукло.

♦ Функция f(x) называется н е п р е р ы в н о й на D в точке x^k, если для $\forall \delta > 0$ $\exists \varepsilon > 0$, такое что для $\forall x \in U(x,\varepsilon)$ $\| f(x) - f(x^k) \| \le \delta$.

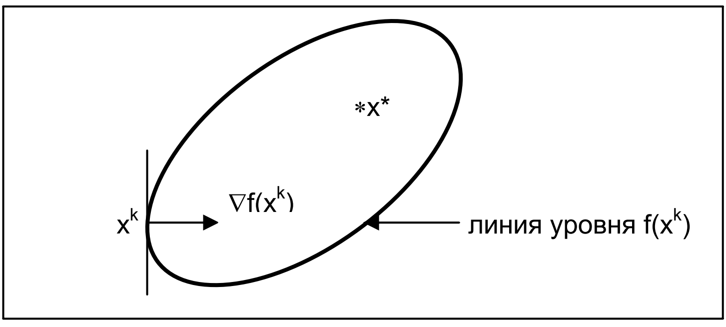
♦ Выпуклая функция f(x), определенная на выпуклом множестве ($x \in D$), является непрерывной в каждой внутренней точке D.

♦ Г р а д и е н т $\nabla f(x^k)$ функции f(x) в точке $x^k \in D$, представляет собой вектор

$$\nabla f(x^k) = \mathrm{grad}\, f(x^k) = \left(\frac{\delta}{\delta x_1} f(x^k), \frac{\delta}{\delta x_2} f(x^k), \ldots, \frac{\delta}{\delta x_n} f(x^k) \right)^{\mathrm{T}} .$$

Вектор градиента показывает направление возрастания значений функции.

Геометрический смысл градиента.



♦ Функция $f(x)$, определенная в окрестности x^k, называется д и ф ф е р е н ц и р у е м о й в x^k, если $\exists \nabla f(x^k)$, такой, что в окрестности точки x^k функция $f(x)$ представима в виде

$$f(x) = f(x^k) + \nabla^{т} f(x^k)(x - x^k) + o\|x - x^k\| \qquad (2.2)$$

♦ Функция $f(x)$ дифференцируема в x^k, если она имеет в окрестности этой точки частные производные $(\delta f/\delta x_i)$, непрерывные в точке x^k.

♦ Функция f принадлежит классу C_o, если она непрерывна.

♦ Функция f принадлежит классу C_1, если она непрерывно дифференцируема - имеет непрерывные частные производные по каждому из своих аргументов. (Называется "C_1- гладкая" или просто "гладкая").

♦ Функция f принадлежит классу $C_к$, если у нее непрерывны все частные производные по каждому из своих аргументов вплоть до к-го порядка.

Анализируя свойства выпуклости множества допустимых решений и целевой функции, можно исследовать существование и единственность решения задачи математического программирования. В частности, существует утверждение, что, если $f(x)$ вогнута (выпукла вниз) на выпуклом, замкнутом и ограниченном множестве Δ, то решение задачи математического программирования (2.1) существует; такие задачи называются задачами выпуклого математического программирования.

2.4. Необходимые и достаточные условия существования оптимального решения. Теорема Куна-Таккера

В задаче математического программирования (2.1)

$$x^* = \arg \max_{x \in \Delta} f(x), \text{ где } \Delta = \{x \in R^n \mid \varphi_i(x) \le b_i,\ i = 1, \ldots, m\}, \qquad (2.3)$$

будем полагать, что $f(x)$-вогнутая (выпуклая вверх) функция, и все функции $\varphi_i(x)$, $i = 1,\ldots, m$ выпуклы. Тогда множества $\varphi_i(x) \le b_i$, $i = 1,\ldots, m$, ограниченные линией уровня выпуклой функции, выпуклы, а, следовательно, и Δ - выпукло, т.к. как является пересечением m выпуклых множеств.

Такая задача является задачей выпуклого программирования, и для нее можно сформулировать необходимые и достаточные условия оптимальности решения [13]. Такие условия задаются теоремой Куна-Таккера, которая представляет собой обобщение классического метода множителей Лагранжа для определения экстремума функции при наличии ограничений.

Для формулирования необходимых и достаточных условий будем рассматривать функцию Лагранжа в виде

$$L(x, \mu) = f(x) + \mu^{T}(b - \varphi(x)) = f(x) + \sum_{i=1}^{m} \mu_i (b_i - \varphi_i(x)), \qquad (2.4)$$

где $\mu_i \geq 0$, $i=1,\ldots, m$ - множители Лагранжа.

Множество Δ является р е г у л я р н ы м, если $\exists x \in \Delta$, такое, что $f(x) < b$, т.е. в Δ существует внутренняя точка (условие Слейтера).

Сформулируем следующую теорему.

Теорема (Куна-Таккера)

Для того, чтобы точка x^* была решением задачи выпуклого программирования (2.3) в условиях регулярности Δ необходимо и достаточно существование такого вектора $\mu^* \geq 0$, чтобы для точки (x^*,μ^*) выполнялись условия

$$L(x, \mu^*) \leq L(x^*, \mu^*) \leq L(x^*, \mu) . \qquad (2.5)$$

Условия (2.5) являются условиями существования седловой точки функции Лагранжа - функции двух векторных аргументов. Точка (x^*, μ^*) называется с е д л о в о й точкой функции Лагранжа $L(x, \mu)$, $x \geq 0$, $\mu \geq 0$.

Поясним данную теорему, для этого подставим (2.4) в (2.5), тогда

$$f(x) + \mu^{*T}(b - \varphi(x)) \leq f(x^*) + \mu^{*T}(b - \varphi(x^*)) \leq f(x^*) + \mu^{T}(b - \varphi(x^*)). \qquad (2.6)$$

Рассмотрим вторую группы неравенств в (2.6), откуда можно заключить, что

$$\mu^{*T}(b - \varphi(x^*)) \leq \mu^{T}(b - \varphi(x^*)).$$

Т.к. $\mu^* = \min \mu$, при $\forall \mu \geq 0$, то отсюда следует, что последнее неравенство может выполняться только в случае, если

$$(b - \varphi(x^*)) \geq 0.$$

Таким образом, точка x^* удовлетворяет всем ограничениям, т.е. она д о п у с т и м а ($x^* \in \Delta$).

Далее, поскольку последнее неравенство должно выполняться при $\forall \mu \geq 0$, и, как было показано, $(b - \varphi(x^*)) \geq 0$, то отсюда непосредственно следует, что

$$\mu^{*T}(b - \varphi(x^*)) = 0. \qquad (2.7)$$

Условия (2.7) называются условиями д о п о л н я ю щ е й н е ж е с т к о с т и [13]. Данные условия устанавливают связь μ^* и x^*. Смысл их в том, что

- если $\mu^*_i = 0$, то $b_i - \varphi_i(x^*)) > 0$ или, если $b_i - \varphi_i(x^*)) > 0$, то $\mu^*_i = 0$;
- если $\mu^*_k > 0$, то $b_k - \varphi_k(x^*)) = 0$ или, если $b_k - \varphi_k(x^*)) = 0$, то $\mu^*_k > 0$.

Рассмотрим первую группу неравенств в (2.6). С учетом условий дополняющей нежесткости (2.7) их можно переписать как

$$f(x) + \mu^{*T}(b - \varphi(x)) \le f(x^*),$$

т.к. $\mu^{*T}(b - \varphi(x)) \ge 0$, $x \in \Delta$, то для $\forall x \in \Delta$ выполняется $f(x) \le f(x^*)$, следовательно, точка x^* - о п т и м а л ь н а на Δ.

Таким образом, если выполняются условия теоремы Куна-Таккера, то решение x^* допустимо и оптимально, т.е. является решением задачи (2.3).

Приведем другую запись теоремы Куна-Таккера. Для этого обозначим

$$L(x, \mu^*) = \min_{\mu} L(x, \mu) = \Phi(x), \text{ тогда } \max_{x} \Phi(x) = L(x^*, \mu^*).$$

Следовательно, $L(x^*, \mu^*) = \max\limits_{x} \min\limits_{\mu} L(x, \mu)$

С другой стороны введем обозначение

$$L(x^*, \mu) = \max_{x} L(x, \mu) = \Psi(\mu), \text{ тогда } \min_{\mu} \Psi(\mu) = L(x^*, \mu^*).$$

Следовательно, $L(x^*, \mu^*) = \min\limits_{\mu} \max\limits_{x} L(x, \mu)$

Тогда

$$\max_{x} \min_{\mu} L(x, \mu) = L(x^*, \mu^*) = \min_{\mu} \max_{x} L(x, \mu)$$

Таким образом, задачи поиска максимума по x и задачи поиска минимума по μ функции Лагранжа $L(x,\mu)$ тесно связаны.

$$\max_{x} \underbrace{\min_{\mu} L(x, \mu)}_{\Phi(x)} = \min_{\mu} \underbrace{\max_{x} L(x, \mu)}_{\Psi(\mu)}$$

Задачи $\max \Phi(x)$ и $\min \Psi(\mu)$ называются д в о й с т в е н н ы м и (сопряженными) задачами и должны решаться совместно. Это положение должно учитываться при построении алгоритмов поиска оптимальных решений.

3. ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

Наиболее изученными и широко применяемыми на практике при принятии различных решений являются модели линейного программирования. Одной из первых практических задач линейного программирования была задача о выборе производственной программы, которая была решена в 1939 г. в работе известного российского математика академика Л.В.Канторовича [44], однако наиболее интенсивное развитие линейное программирование получило в период 1955-1965 г.г. в связи с работами Данцига Дж. [26], Гольштейна Е.Г., Юдина Д.Б. [87]. В настоящее время имеются значительные успехи в части создания эффективных алгоритмов решения этих задач, кроме того, ряд более сложных процедур оптимизации строится на основе использования методов линейного программирования. В связи с этим овладение методикой построения моделей, глубокое понимание сущности алгоритмов линейного программирования является первым шагом на пути практического использования теории оптимизации для принятия решений во всем ее многообразии.

3.1. Постановка задач линейного программирования

Задача математического программирования называется задачей л и н е й н о г о программирования, если целевая функция и ограничения задаются линейными функциями.

Таким образом, задача линейного программирования (ЛП) имеет вид

$$x^* = \arg \max_{x \in \Delta} \sum_{j=1}^{n} c_j\, x_j\,, \ \ \Delta = \{x \in R_+^n \mid \sum_{j=1}^{n} a_{ij}\, x_j \le b_i,\ i = 1,\ldots,m\}$$

или

$$\sum_{j=1}^{n} c_j\ x_j \to \max \qquad\qquad (c,x) = c^т x \to \max \qquad (3.1)$$

$$\sum_{j=1}^{n} a_{ij}\ x_j \le b_i,\ i = 1,\dots, m \qquad \text{или} \qquad A\,x \le b \qquad (3.2)$$

$$x_j \ge 0,\ \ j = 1,\dots, n \qquad\qquad x \ge 0 \qquad (3.3)$$

Это с т а н д а р т н а я форма записи задачи линейного программирования.

Терминология задач линейного программирования

Рассмотрим задачу линейного программирования в стандартной постановке

$$L = c^т x \to \max$$
$$A\,x \le b,$$
$$x \ge 0$$

Здесь

L - значение целевой функции (линейной формы);

A - матрица условий (ограничений);

b - вектор ограничений;

x - допустимое решение (план), если оно удовлетворяет ограничениям (3.2), (3.3);

c - вектор коэффициентов целевой функции (линейной формы).

Матрицу A можно представить в виде совокупности векторов условий

$A = \| a_1, a_2, \dots, a_j, \dots, a_n \|$,

тогда a_j – j-ый вектор условий; или в виде совокупности векторов ограничений

$$A = \left\| \begin{matrix} a^т{}_1 \\ \dots \\ a^т{}_i \\ \dots \\ a^т{}_m \end{matrix} \right\|$$

здесь $a^т{}_i = \| a_{i1}, a_{i2}, \dots, a_{ij}, \dots, a_{in} \|$ - i-ый вектор ограничений задачи линейного программирования; i = 1,..., m.

Двойственные задачи линейного программирования

Для задачи линейного программирования (3.1)-(3.3) функция Лагранжа имеет вид

$$L(x, \mu) = c^T x + \mu^T(b - Ax) = c^T x + \mu^T b - \mu^T Ax = \mu^T b + (c^T - \mu^T A)x = \ldots$$

т.к. $L(x,\mu)$ - число, то последнее выражение можно транспонировать, тогда

$$\ldots = b^T\mu + x^T(c - A^T\mu)$$

Итак, с одной стороны $L(x, \mu) = c^T x + \mu^T(b - Ax)$ (3.4)

с другой стороны $L(x, \mu) = b^T\mu + x^T(c - A^T\mu)$ (3.5)

Из теоремы Куна-Таккера (см. 2.4) следует, что

$$\max_x \min_\mu L(x, \mu) = \min_\mu \max_x L(x, \mu)$$

или с учетом (3.4),(3.5)

$$\max_x \underbrace{\min_\mu (c^T x + \mu^T(b - Ax))}_{\Phi(x)} = \min_\mu \underbrace{\max_x (b^T\mu + x^T(c - A^T\mu))}_{\Psi(\mu)}$$

Последнее выражение описывает связь двойственных задач, его можно переписать в виде

$$\max_x (c^T x + \min_\mu \mu^T(b - Ax)) = \min_\mu (b^T\mu + \max_x x^T(c - A^T\mu))$$

Здесь $\min \mu^T(b - Ax)$, при любом $\mu \geq 0$ не существует, если $(b - Ax) \leq 0$. Следовательно, должно выполняться условие $(b - Ax) \geq 0$.

Аналогично, $\max x^T(c - A^T\mu)$, при любом $x \geq 0$ не существует, если $(c - A^T\mu) \geq 0$. Следовательно, должно выполняться условие $(c - A^T\mu)) \leq 0$.

Отсюда получаем пару д в о й с т в е н н ы х (сопряженных) задач, решения которых тесно связаны друг с другом:

прямая задача	**двойственная задача**
$L = c^T x \to \max$	$M = b^T\mu \to \min$
$A x \leq b,$	$A^T \mu \geq c$
$x \geq 0$	$\mu \geq 0$

Можно показать, что если рассматривать двойственную задачу, как прямую, то прямая задача будет по отношению к ней двойственной.

Вектор ограничений b часто интерпретируется как вектор ресурсов, необходимых для реализации решения x. Тогда двойственные переменные μ можно интерпретировать, как стоимость (ценность) использования ресурсов в единицах критерия L.

Рассмотрим свойства двойственных задач линейного программирования.

1). Значение целевой функции прямой задачи, вычисленное для некоторого решения x, не превышает значения целевой функции двойственной задачи.

$$c^{T}x \le b^{T}\mu$$

2). Если x^* и μ^* - оптимальные решения прямой и двойственной задач, то $c^{T}x^* = b^{T}\mu^*$ (то есть L = M).

3). Если в прямой задаче $\sum a_{ij}\, x_j^* < b_i$, то $\mu_i^* = 0$. Действительно, если i-го ресурса в избытке, то стоимость его использования равна 0 .

4). Если в двойственной задаче $\sum a_{ij}\, \mu_j^* > c_j$, то $x_i^* = 0$. Действительно, если затраты при использовании j-го ресурса превышают прибыль, то такой ресурс нет смысла использовать.

Отметим, что свойства 1, 2 вытекают непосредственно из теоремы Куна-Таккера; свойства 3, 4 являются интерпретацией условий дополняющей нежесткости для задач линейного программирования (см. 2.4).

<u>Каноническая форма задач линейного программирования</u>

Наряду со стандартной формой записи задач линейного программирования (3.1)-(3.3) широко используется запись задач в канонической форме.

$$L = c^{T}x \to \max$$
$$A\,x = b, \tag{3.6}$$
$$x \ge 0$$

при этом $b \ge 0$.

Каноническая форма записи является более общей, чем стандартная, так как произвольная задача линейного программирования может быть сведена к канонической форме. Действительно, пусть имеется s-е ограничение вида

$$\sum_{j=1}^{n} a_{sj}x_j \le b_s\,,$$

тогда в канонической форме оно будет иметь следующий вид

$$\sum_{j=1}^{n} a_{sj}x_j + x_{n+1} = b_s$$

Значение переменной x_{n+1} характеризует величину, на которую левая часть меньше правой. В целевой функции переменной x_{n+1} соответствует 0, т.е. она имеет вид $(c_1,...,c_n,0)\ x \to \max$. Соответственно, если k-е ограничение имеет вид

$\sum_{j=1}^{n} a_{kj}x_j \geq b_k$, тогда в канонической форме $\sum_{j=1}^{n} a_{kj}x_j - x_{n+1} = b_k$

Задача, двойственная к прямой задаче линейного программирования в канонической форме, имеет вид

$$L = b^{т}\mu \to \min$$
$$A^{т}\mu \geq c$$

Т.е. компоненты вектора μ могут принимать и отрицательные значения. Действительно, стандартная задача линейного программирования (3.1)-(3.3) в канонической форме будет иметь вид

$$c^{т}x + 0^{т}x_d \to \max$$
$$A\,x + E\,x_d = b,$$
$$x \geq 0,\ x_d \geq 0.$$

Здесь x_d - дополнительные переменные;

E - единичная матрица;

$0^{т}$ – транспонированный вектор, все компоненты которого равны 0.

Задача двойственная к последней будет иметь вид

$$b^{т}\mu \to \min$$
$$A^{т}\mu .\geq c,$$
$$E\,\mu \geq.\ 0,\ откуда\ \mu \geq 0$$

т.е. будет иметь вид задачи, двойственной к задаче линейного программирования в стандартной форме.

3.2. Свойства выпуклых многогранников в задаче линейного программирования

Пусть задача линейного программирования поставлена в стандартной форме (3.1)-(3.3)

$$L = c^{т}x \to \max$$
$$A\,x \leq b\ .$$
$$x \geq 0$$

или

$$
\begin{aligned}
&L = c_1x_1 + \ldots + c_nx_n \to \max\\
&\quad a_{11}x_1 + \ldots + a_{1n}x_n \le b_1,\\
&\quad \ldots\ldots\ldots\ldots\ldots\ldots\\
&\quad a_{m1}x_1 + \ldots + a_{mn}x_n \le b_m,\\
&\qquad x_j \ge 0,\ j = 1,\ldots, n
\end{aligned}
$$

Каждое i-е ограничение задачи линейного программирования задает некоторое замкнутое полупространство.

Сформулируем ряд определений и утверждений, которые будут использоваться в дальнейшем.

Выпуклые многогранники

♦ В ы п у к л о е м н о г о г р а н н о е м н о ж е с т в о - непустое пересечение конечного числа замкнутых полупространств.

♦ В ы п у к л ы й м н о г о г р а н н и к - ограниченное многогранное множество.

♦ Г р а н ь ю многогранного множества размерности h в R^n называется пересечение n-h линейно независимых гиперплоскостей.

♦ В е р ш и н о й многогранника называется грань нулевой размерности (пересечение n линейно независимых гиперплоскостей).

♦ Вершина многогранника M - крайняя точка M.

Линейная независимость

♦ Система векторов $x_1,\ldots,x_r \in D \subseteq R^n$ называется л и н е й н о з а в и с и м о й, если в линейной комбинации векторов

$$\sum_{i=1}^{r} \alpha_i x_i = 0 \text{ существует } k \in \{1,\ldots,r\}, \text{ такое, что } \alpha_k \ne 0$$

В противном случае система векторов называется л и н е й н о н е з а в и с и м о й. Т.е., если

$$\sum_{i=1}^{r} \alpha_i x_i = 0 \text{ , то } \forall \alpha_i = 0 .$$

♦ Если система векторов $\{x_1,\ldots, x_r\}$ линейно зависима, то хотя бы один из векторов является линейной комбинацией остальных, т.е. $\exists x_k \mid x_k = \sum_{i=1, i\neq k}^{r} \alpha_i x_i$.

♦ Линейно независимая система векторов называется м а к с и м а л ь н о й линейно независимой системой в D, если добавление к ней любого вектора из D делает ее линейно зависимой.

♦ Любая максимальная линейно независимая система векторов называется б а з и с о м (базой) в R^n.

♦ Р а н г о м системы векторов $D = \{x_1,..., x_q\}$ называется количество векторов в максимальной линейно независимой системе; rang D = r.

♦ Ранг матрицы A (множество векторов) равен максимальному порядку отличных от нуля миноров A.

♦ Максимальное число линейно независимых строк A равно максимальному числу линейно независимых столбцов и равно рангу A.

Утверждение.

Пусть задан многогранник M(A, b) ограничениями $Ax \leq b$, $x \geq 0$, где A - матрица размерности $m \times n$. Тогда, если решение x соответствует вершине многогранника M(A, b), то оно называется о п о р н ы м решением и представимо в виде

$$x = (x_s, x_q)^T,$$

где x_s - базисные переменные x_q - небазисные (свободные) переменные. Причем,

$$x_s \geq 0,\ |x_s| = m;\ x_q = 0,\ |x_q| = n\text{-}m,$$

т.е. опорное решение представимо в виде $x = (x_s, 0)^T$.

Тогда $A = \| S \mid Q \|$,

где $S = \| a_{s1}, a_{s2}, \ldots ,a_{sm} \|$ - матрица базиса и множество векторов $\{a_{s1}, a_{s2}, \ldots ,a_{sm}\}$ - базис матрицы A.

$J_s = \{s1,s2,\ldots,sm\}$ - номера векторов условий, образующих базис.

Если x - опорное решение, то ограничения задачи линейного программирования могут быть представлены в виде

$$A x = b \Rightarrow S x_s + Q x_q = b,$$

откуда следует, что $x_s = S^{-1} b$, т.к. $x_q = 0$.

Действительно, в любой вершине многогранника M(A,b) пересекаются n гиперплоскостей. Причем m гиперплоскостей задаются ограничениями вида $A x = b$, а (n-m) гиперплоскостей задаются ограничениями вида $x_j = 0$

<u>Утверждение.</u>

Оптимальное решение задачи линейного программирования соответствует хотя бы одной вершине (крайней точке) выпуклого многогранника M(A,b).

Действительно, пусть известно множество крайних точек M(A,b), а именно $\{x^*_1, x^*_2, \ldots, x^*_s\}$. Тогда $\forall x \in M(A,b)$ представимо в виде $x = \sum_i \alpha_i x^*_i$. Тогда

$$c^T x = c^T \sum_i \alpha_i x^*_i = \sum_i \alpha_i c^T x^*_i = \sum_i \alpha_i L^*_i \to \max$$

$L^*_k = \max_i L^*_i$, т.е. оптимум достигается в к-ой крайней точке (вершине многогранника).

Очевидно, что число вершин выпуклого многогранника, задаваемого ограничениями задачи линейного программирования, конечно. Тогда, на основе организации направленного перебора решений x, соответствующих вершинам многогранника (опорных решений), и, сравнения значений целевой функции, вычисленных при этих решениях, можно найти оптимальное решение x^*, соответствующее максимальному значению целевой функции.

3.3. Геометрическое представление задач линейного программирования

Задача линейного программирования в стандартной форме имеет вид:

$$L = c^T x \to \max$$
$$A x \le b \,.$$
$$x \ge 0$$

Здесь $x \in R^n$, пусть n = 2. Тогда задача будет иметь вид

$$c_1 x_1 + c_2 x_2 \to \max \qquad (3.7)$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 &\le b_1 \\ &\ldots\ldots\ldots\ldots \\ a_{m1} x_1 + a_{m2} x_2 &\le b_m \end{aligned} \qquad (3.8)$$

$$x_1, x_2 \ge 0 \qquad (3.9)$$

Структура области допустимых решений

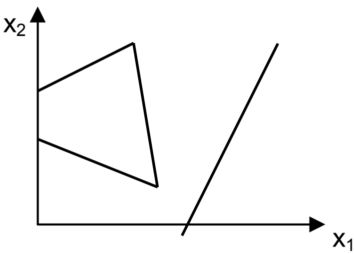
♦ Ограничения (3.9) задают положительный ортант (2 полупространства).

♦ Ограничения (3.8) задают пересечение m полупространств в R^n.

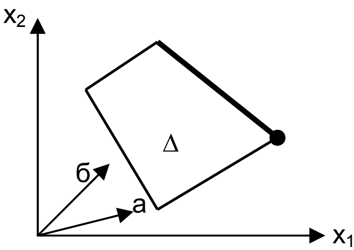
♦ $\{x^*_1, x^*_2, \ldots, x^*_s\}$ - вершины выпуклого многогранника (0-мерные грани), в вершинах пересекаются n (n=2) гиперплоскостей.

♦ $\forall x \in M(A,b)$ - линейная комбинация крайних точек x^*_i. Причем для представления любого x таких точек должно быть не более 3.

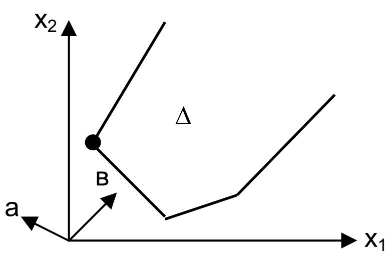
♦ В зависимости от структуры ограничений (3.8), (3.9) можно различать 3 случая:



А).
$\Delta = \varnothing$ - ограничения противоречивы в совокупности.



Б).
$\Delta \neq \varnothing$ и Δ - ограниченное множество (т.е. Δ- многогранник).



В).
$\Delta \neq \varnothing$ и Δ - неограниченное множество (Δ - многогранное множество).

Целевая функция.

Целевая функция в задаче линейного программирования имеет вид

$$L = c_1 x_1 + c_2 x_2$$

Тогда $L_1 = c_1 x_1 + c_2 x_2$ - линия уровня L_1 (гиперплоскость). Градиент целевой функции

$$\nabla L = \left\| \frac{\partial L}{\partial x_i} \right\| = (c_1 \ c_2)^T$$

Линия уровня перпендикулярна градиенту ∇L.

В зависимости от направленности градиента можно различать ситуации:

- в случае А) - нет ни одного решения (не зависимо от направлен ности градиента).
- в случае Б): а - решение существует и оно единственно (решение соответствует некоторой вершине многогранника) ;
  б - решение существует и оно не единственно, существует множество эквивалентных решений (решение соответствует некоторой грани многогранника) ;
- в случае В): а - решение существует и оно единственно;
  б - решение существует и оно не единственно;
  в - решения не существует - целевая функция неограниченно возрастает на множестве допустимых решений;

Таким образом, можно сформулировать следующие свойства решений задачи линейного программирования.

Свойства решений задачи линейного программирования

♦ Оптимальное решение, если оно существует, находится на границе допустимой области.

♦ Оптимальное решение всегда достигается в некоторой крайней точке (вершине) многогранника M(A,b).

♦ Оптимальное решение может быть не единственным, тогда оно достигается как минимум в 2-х крайних точках.

♦ Задача может не иметь решения, хотя допустимые решения и существуют (целевая функция не ограничена на множестве допустимых решений).

3.4. Основные элементы алгоритмов линейного программирования

Пусть задача линейного программирования задается в каноническом виде:

$$L = c^T x \to \max \qquad c_1 x_1 + \ldots + c_n x_n \to \max$$

$$A x = b \quad \text{или} \quad \begin{array}{l} a_{11}x_1 + \ldots + a_{1n}x_n = b_1 \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + \ldots + a_{mn} x_n = b_m \end{array}$$

$$x \geq 0 \qquad x_j \geq 0,\ j = 1,\ldots, n$$

или

$$\| c_1\ c_2 ,\ldots, c_n \| x \to \max$$

$$\left\| \begin{array}{c} a_{11}\ a_{12}\ ,\ldots,\ a_{1n} \\ \ldots\ldots\ldots \\ a_{m1}\ a_{m2}\ ,\ldots,\ a_{mn} \end{array} \right\| \times \left\| \begin{array}{c} x_1 \\ \ldots \\ x_m \end{array} \right\| = \left\| \begin{array}{c} b_1 \\ \ldots \\ b_m \end{array} \right\|$$

Матрицу условий можно представить в виде совокупности векторов условий

$$A = \| a_1\ a_2 ,\ldots, a_n \| .$$

Здесь a_j, $j=1,\ldots,n$ - векторы условий задачи линейного программирования.

Пусть $J = \{1,\ldots, n\}$ - множество номеров векторов условий. Тогда можно выделить $J_s = \{s1,\ldots,sm\} \subset J$, такое, что множество векторов условий $\{a_{s1}, a_{s2},\ldots, a_{sm}\}$ образуют базис матрицы A (максимальную линейно независимую систему векторов).

$S = \| a_{s1}\ a_{s2} ,\ldots, a_{sm} \| = \| a_j ,\ j \in J_s \|$ - матрица базиса.

Если $x_s = S^{-1} b$, такое, что $x_s \geq 0$, то

S называется матрицей о п о р н о г о базиса, а

x_s называется о п о р н ы м решением.

Опорное решение соответствует вершине многогранника решений задачи линейного программирования.

Множество $J_q = J \setminus J_s$ - множество номеров небазисных (свободных) переменных. Тогда $x = \| x_s\ x_q \|^T$, где x_s - базисные переменные; x_q - небазисные (свободные) переменные;

$x_s \geq 0$; $| x_s | = m$; $x_q = 0$; $| x_q | = n - m$.

Тогда матрицу A можно представить в виде

$$A = \| S\ Q \| ,$$

где $S = \| a_j ,\ j \in J_s \|$; $Q = \| a_j ,\ j \in J_q = J \setminus J_s \|$.

Решение $x_s = S^{-1} b$ - опорное решение; $c_s = \| c_j ,\ j \in J_s \|$ - компоненты вектора коэффициентов целевой функции, соответствующие опорному решению, $L(x_s) = c^T_s x_s$.

Рассмотрим основные элементы, составляющие формальную основу различных алгоритмов линейного программирования (в частности, симплекс-метода).

1. Признак оптимальности опорного решения (плана)

Признаком оптимальности опорного решения выступает следующее условие

$$\pi^т A - c^т \geq 0 \quad \text{или} \quad \pi^т a_j - c_j \geq 0, \; j = 1,.., n \qquad (3.10)$$

где

$$\pi^т = c^т{}_s S^{-1} \qquad (3.11)$$

Здесь π - вектор двойственных оценок векторов условий относительно текущего базиса (разрешающий вектор).

Покажем, что если выполняется данный признак, то опорное решение оптимально. Действительно,

$$\pi^т A - c^т \geq 0, \; \text{откуда} \quad \pi^т A\,x \geq c^т x = L(x), \; \forall x \in \Delta$$

В то же время левая часть последнего неравенства

$$\pi^т A\,x = c^т{}_s S^{-1} A\,x = c^т{}_s S^{-1} b = c^т{}_s x_s = L(x_s)$$

Т.е. $L(x_s) \geq L(x)$, $\forall x \in \Delta$, следовательно, x_s - оптимально ($x_s = x^*{}_s$).

С другой стороны, если $\exists k \in J \mid \pi^т a_k - c_k < 0$, то $\exists x \in \Delta \mid L(x) \geq L(x_s)$, т.е. существует другая вершина многогранника условий с большим значением целевой функции.

2). Приращение целевой функции

Итак, если признак оптимальности не выполняется, то существует другое опорное решение, позволяющее увеличить значение целевой функции. Здесь можно сформулировать два вопроса:

- на какую величину возрастет значение целевой функции;
- каким образом этого увеличения достичь.

Если признак оптимальности не выполняется, следовательно, хотя бы одну из небазисных переменных целесообразно ввести в базис, т.е. она должна принять значение большее 0. Пусть это будет k-я переменная x_k, и она примет значение θ, т.е.

$$x_k = \theta > 0.$$

Тогда, если

$$S\,x_s = b,$$

то $\quad S\,x_s - a_k x_k + a_k x_k = b,$

или $\quad S\,x_s - \theta\,a_k + \theta\,a_k = b,$

или $\quad S(x_s - \theta\,S^{-1} a_k) + \theta\,a_k = b,$

или, обозначая $æ_k = S^{-1}a_k$,

$$S(x_s - \theta æ_k) + \theta\, a_k = b.$$

Обозначим $x_s(\theta) = x_s - \theta æ_k$,

тогда $S\, x_s(\theta) + \theta\, a_k = b$.

Значение целевой функции, вычисленное для $x_s(\theta)$

$$\begin{aligned}
L(x_s(\theta)) &= c^T{}_s x_s(\theta) + c_k\, \theta = \\
&= c^T{}_s(x_s - \theta\, æ_k) + c_k\, \theta = \\
&= c^T{}_s x_s - c^T{}_s\, \theta\, æ_k + c_k\, \theta = \dots \text{ т.к. } æ_k = S^{-1}a_k \\
&= c^T{}_s x_s - \theta\, (\, c^T{}_s\, S^{-1}\, a_k - c_k) = \\
&= L(x_s) - \theta\, \delta_k,
\end{aligned}$$

где $\delta_k = c^T{}_s\, S^{-1}\, a_k - c_k = \pi^T\, a_k - c_k$.

Тогда приращение значения целевой функции

$$\Delta L = L(x_s(\theta)) - L(x_s) = -\,\theta\, \delta_k \qquad (3.12)$$

Следовательно, для того чтобы приращения целевой функции были по возможности большими δ_k необходимо выбирать из условия

$$\delta_k = \min_j\, (\pi^T\, a_j - c_j),\ \ j \in J_q \qquad (3.13)$$

<u>3). Замена векторов базиса</u>

Значение переменной x_k, которая становится базисной (вводится в базис) равно $x_k = \theta$. Однако в опорном решении может быть только m отличных от 0 переменных, и если x_k становится равной θ, то одна из базисных переменных должна стать равной 0, или некоторый r-ый вектор базиса S должен быть выведен из него. Возникает вопрос - какой из векторов базиса следует вывести из базиса для того, чтобы на его место ввести вектор a_k?

Пусть из базиса выводится некоторый r-ый вектор, r = 1,..., m. Следовательно, r-я компонента вектора $x_s(\theta)$ должна стать равной 0.

В целом новое опорное решение определяется как

$$x_s(\theta) = \begin{cases} x'_{si} = x_{si} - \theta\, æ_{ki},\ i = 1,..,m,\ i \neq r, \\ \\ x'_{sr} = \theta \end{cases} \qquad (3.14)$$

Отметим, что если $\forall æ_{ki} < 0$, то θ, а, следовательно, и значение целевой функции L и компоненты x_s могут принимать любые сколь угодно большие значения. Таким образом, если все $æ_{ki} < 0$, то целевая функция может неограниченно возрастать на множестве допустимых решений задачи линейного программирования. Следовательно, оптимального решения не существует.

Все компоненты x_s должны быть больше 0, т.е.

$$x'_{si} = x_{si} - \theta\, æ_{ki} > 0, \quad i = 1,..,m,$$

Это выполняется, если выбирать θ из условия

$$\theta = \min_{æ_{ki}>0} \frac{x_{si}}{æ_{ki}}, \quad i = 1,..,m,$$

причем по тем компонентам $æ_{ki}$, которые больше 0; $æ_{ki} > 0$.

Тогда из базиса выводится вектор с номером r, который определяется из условия:

$$\theta_r = \min_{æ_{ki}>0} \frac{x_{si}}{æ_{ki}}, \quad i = 1,..,m. \tag{3.15}$$

Производится замена векторов базиса по правилу

$$J'_s = J_s \cup \{k\} \setminus J_s(r) \tag{3.16}$$

Здесь $J_s(r)$ - номер переменной находящейся на r-ой позиции в базисе.

3.5. Алгоритмы линейного программирования. Симплекс-метод

Алгоритмы решения задачи линейного программирования можно разбить на два класса. Алгоритмы первого класса основываются на конечных методах, обеспечивающих решение задачи за конечное число шагов. Алгоритмы второго класса основываются на итеративных методах, позволяющих получить приближенное решение задачи и связанных с проведением достаточно большого числа итераций. При этом качество приближения существенно зависит от числа проведенных итераций.

Конечные методы решения задач линейного программирования можно разделить на следующие три группы [87]:

1. Методы последовательного улучшения решений, обеспечивающие получение оптимального решения задачи при движении по планам прямой задачи.

2. Методы последовательного уточнения оценок, в которых оптимальное решение достигается при движении по планам двойственной (сопряженной) задачи.

3. Методы последовательного сокращения невязок, в которых используются обе задачи двойственной пары.

В настоящее время методы второй и третьей группы используются в специальных приложениях (целочисленное программирование, приближенное решение задач и др.), в то время как практические алгоритмы решения общей задачи линейного программирования строятся на основе методов первой группы, причем в методах первой группы различают: табличный метод и метод обратной матрицы. Различают модификации метода обратной матрицы: метод с мультипликативным представлением обратной матрицы (мультипликативный алгоритм) и метод с треугольным (верхним и нижним) разложением обратной матрицы.

Первоначально симплекс-методом назывался именно табличный метод, в то время как метод обратной матрицы назывался модифицированным симплекс-методом или вторым алгоритмом симплекс-метода. Вместе с тем метод обратной матрицы по сравнению с табличным характеризуется меньшим объемом вычислений, более компактным представлением необходимых для решения задачи данных, возможностью получения наряду с решением прямой задачи решения двойственной задачи. Именно поэтому реальные алгоритмы линейного программирования строятся на основе этого метода, и он обладает большим числом различных модификаций. В дальнейшем, говоря о симплекс-методе, будем иметь в виду именно метод обратной матрицы.

Так как решение задачи линейного программирования соответствует вершине многогранника, оно определяется опорным базисом J_s и соответствующей ему матрицей опорного базиса S. Тогда оптимальное решение может быть найдено перебором различных квадратных подматриц S матрицы A и проверкой условий, которым должна удовлетворять матрица опорного базиса:

- $\det S \neq 0$;
- $x_s = S^{-1} b \geq 0$.

Всего количество таких матриц S, которые таким образом можно составить, не превышает числа сочетаний C^m_n (здесь m и n размерность матрицы условий A). Каждой матрице S соответствует свое решение x_s и значение целевой функции $L(x_s)$. Сравнивая значения $L(x_s)$, можно выявить решение x^*_s с наибольшим значением целевой функции $L(x^*_s)$.

Симплекс-метод осуществляет направленный перебор таких матриц S, соответствующих вершинам многогранника решений, причем каждый новый выбор S характеризуется улучшением значения целевой функции. Так как число вершин многогранника решений конечно, то за конечное число шагов достигается оптимальное решение x^* ($L(x^*) = max$) или устанавливается неразрешимость задачи.

Алгоритм симплекс-метода

Алгоритм представляет собой определенную итеративную последовательность действий, связанную с нахождением оптимального решения. В соответствии с этим на каждой итерации выполняется три шага:

♦ анализ оптимальности текущего решения; нахождение вектора, вводимого в базис;

♦ установление неразрешимости задачи; нахождение вектора, выводимого из базиса;

♦ преобразование базиса.

Все данные, необходимые для проведения расчетов на каждой итерации, целесообразно представлять в форме таблицы следующего вида:

Таблица 1.

J_s	L	$\pi_1\ \pi_2\ \ldots\ldots\ \pi_m$	δ_k	δ_k
s1	x_{s1}	$s_{11}\ s_{12}\ \ldots\ s_{1m}$	$æ_{k1}$	a_{k1}
s2	x_{s2}	$s_{21}\ s_{22}\ \ldots\ s_{2m}$	$æ_{k2}$	a_{k2}
….	….	………………	….	….
….	….	S^{-1}	….	….
….	….	………………	….	….
sm	x_{sm}	$s_{m1}\ s_{m2}\ \ldots\ s_{mm}$	$æ_{km}$	a_{km}

Часть таблицы, обведенную двойной линией, называют о с н о в н о й ч а с т ь ю таблицы 1.

Предположение.

Пусть имеется некоторое опорное решение. Тогда известны

- $J_s = \{s1, s2, \ldots, sm\}$ множество номеров векторов условий базиса;

- $S = \| a_{s1}\ \ a_{s2}\ , \ldots, a_{sm} \| = \| a_j\ ,\ j \in J_s \|$ - матрица базиса; соответственно известны элементы матрицы $S^{-1} = \| s_{ij} \|$;

- $x_s = S^{-1}\, b$, такое, что $x_s \geq 0$;

- $\pi^T = c^T_s S^{-1}$;
- $L(x_s) = c^T_s x_s$.

Таким образом, если известно опорное решение, то можно заполнить основную часть таблицы 1.

1. Анализ оптимальности текущего решения; нахождение вектора, вводимого в базис

В соответствии с (3.13) рассчитывается оценка

$\delta_k = \min (\pi^T a_j - c_j), j = 1,...,m.$

Если $\delta_k \geq 0$ - счет окончен. Текущее опорное решение оптимально.

Если $\delta_k < 0$ - вектор условий a_k целесообразно ввести в базис - производится заполнение крайнего правого столбца таблицы 1.

2. Установление неразрешимости задачи; нахождение вектора, выводимого из базиса

Вектор a_k приводится к текущему базису - находится вектор $æ_k = S^{-1} a_k$. Заполняется второй справа столбец таблицы 1.

Если $æ_{ki} \leq 0$, i=1,...,m - задача не имеет решений (целевая функция может неограниченно возрастать множестве допустимых решений). Счет окончен.

Если $\exists æ_{ki} > 0$, i=1,...,m , то производится вычисление оценки θ_r по формуле (3.15):

$$\theta_r = \min_{æ_{ki}>0} \frac{x_{si}}{æ_{ki}}, \quad i = 1,..,m.$$

Вектор условий, соответствующий r-ой позиции базиса следует вывести из базиса.

3. Преобразование базиса

Итак, r-ый вектор базиса меняется на k-ый вектор условий. Тогда новый базис $J'_s = J_s \cup \{k\} \setminus J_s(r)$, изменяется крайний слева столбец таблицы 1.

Неизменной осталась основная часть таблицы, (обведенная двойной линией) она изменяется в соответствии с формулами:

$$s'_{rj} = \frac{s_{rj}}{æ_{kr}}, \; j = 0,\dots, m, \tag{3.16}$$

$$s'_{ij} = s_{ij} - s'_{rj}\, æ_{ki}, \; i = 0,\dots, m, \; j = 0,\dots, m, \tag{3.17}$$

Таким образом, вначале преобразуется r-я строка основной таблицы, а затем все остальные. За конечное число шагов, заключающихся в выполнении пунктов 1-3, находится оптимальное решение или устанавливается неразрешимость задачи (неограниченность целевой функции на множестве допустимых решений).

Отметим, что алгоритм рассматривался в предположении, что существует некоторое исходное опорное решение, и оно известно (следовательно, $\Delta \neq \varnothing$)

3.6. Построение исходного опорного решения

При рассмотрении алгоритма симплекс-метода вводилось предположение о том, что известно некоторое опорное решение. Вместе с тем задача поиска такого решения является достаточно не тривиальной. Рассмотрим несколько способов построения исходного опорного решения.

<u>Задачи с "очевидным" исходным опорным решением</u>

В ряде случаев задача линейного программирования обладает такими особенностями, что исходное опорное решение может быть найдено, вследствие предварительного анализа задачи. Напомним, что к опорному решению и опорному базису (в том числе и к исходному) предъявляется два требования:

1. Матрица S, составленная из векторов базиса $S = \| a_{s1} \;\; a_{s2}, \dots, a_{sm} \| = \| a_j, \, j \in J_s \|$ является неособенной ($\det S \neq 0$).
2. Компоненты вектора базисных переменных больше 0; $x_s \geq .0$, ($x_s = S^{-1}b$).

Пусть задача линейного программирования задана в стандартной постановке

$$c^{т}x \to \max,$$
$$A\,x \leq b,$$
$$x \geq 0.$$

При сведении такой задачи к канонической форме вводятся дополнительные переменные, и задача приобретает вид

$c^т x + 0^т x_d \to \max,$
$A x + E x_d = b,$
$x \geq 0, \; x_d \geq 0.$

Здесь x_d - дополнительные переменные; E - единичная матрица.

В качестве исходного опорного решения такой задачи можно выбрать дополнительные переменные, т.е.

- $J_s = \{s1, s2, ..., sm\} = \{n+1, n+2, ..., n+m\}$;
- $S = \| e_1 \; e_2, \ldots, e_m \|$, здесь e_j - j-ый вектор матрицы E;
- $x_s = x_d = S^{-1} b = b$;
- $\pi^т = 0^т S^{-1} = 0^т$;
- $L(x_s) = 0^т_s x_s = 0$.

Говорят, что задача линейного программирования в стандартной постановке имеет о ч е в и д н о е опорное решение.

Построение исходного опорного решения на основе вспомогательной задачи

Более общей формой задач линейного программирования является каноническая форма. Здесь для нахождения исходного опорного решения может быть использована вспомогательная задача, которая ставится в виде

$$-1^т x_d \to \max, \tag{3.18}$$
$$A x + E x_d = b, \tag{3.19}$$
$$x \geq 0, \; x_d \geq 0. \tag{3.20}$$

или

$$- x_{n+1} - , \ldots, - x_{n+m} \to \max$$
$$a_{11}x_1 + \ldots + a_{1n}x_n + x_{n+1} \qquad = b_1$$
$$. \; .$$
$$a_{m1}x_1 + \ldots + a_{mn} x_n + \; . \; . \; . \; . \; . \; . \; . + x_{n+m} = b_m$$
$$x_j \geq 0, \; j = 1, \ldots, n+m$$

Такая задача имеет очевидное исходное опорное решение:

- $J_s = \{n+1, n+2, ..., n+m\}$;
- $S = \| e_1 \; e_2, \ldots, e_m \| = E$;
- $x_s = S^{-1} b = b$;
- $\pi^т = -1^т S^{-1} = -1^т = (-1, ..., -1)$;
- $L(x_s) = -1^т x_s = - \Sigma b_i$.

Поскольку целевая функция ограничена сверху на множестве решений такой вспомогательной задачи ($L(x) \leq 0$), то итеративный процесс симплекс-метода через конечное число шагов придет к оптимальному решению вспомогательной задачи. Здесь возможны два случая:

1) оптимальное значение целевой функции $L^* < 0$;
2) оптимальное значение целевой функции $L^* = 0$.

Рассмотрим эти случаи.

Случай 1. При $L^* < 0$ исходная задача линейного программирования не имеет ни одного допустимого решения, то есть система ограничений исходной задачи несовместна ($\Delta = \varnothing$).

Если допустить противное и принять, что исходная задача имеет хотя бы одно допустимое решение, то вспомогательная задача будет иметь решение, у которого первые (n) переменных (основных переменных) совпадают с решением исходной задачи, а остальные (m) переменных (дополнительных переменных) равны 0. Следовательно, оптимальное значение целевой функции вспомогательной задачи с учетом того, что в нее входят только дополнительные переменные (см. 3.18) $L^* = 0$, а это противоречит исходному предположению случая 1, по которому $L^* < 0$.

Случай 2. ($L^* = 0$). Здесь оптимальное решение вспомогательной задачи является опорным решением исходной задачи.

Возможны ситуации, когда полученное опорное решение содержит дополнительные переменные вспомогательной задачи (значения этих переменных, естественно, равны 0).

Рассмотрим эти ситуации более подробно.

Известно, что ранг матрицы условий задачи линейного программирования не превышает числа ограничений. Т.е. возможны две ситуации:

а) $r = \text{rang } A = m$;
б) $r = \text{rang } A < m$.

В ситуации а) дополнительные переменные вспомогательной задачи можно заменить на основные переменные.

В ситуации б) (m-r) ограничений задачи линейного программирования являются линейной комбинацией остальных r ограничений и, следовательно, эти (m-r) ограничений можно исключить, сократив, таким образом, размерность задачи.

Квалифицировать ситуацию (а или б) можно следующим образом.

1). Определяются компоненты разложения векторов условий исходной задачи по оптимальному базису вспомогательной задачи, соответствующие дополнительным переменным. Пусть на p-ой позиции базиса содержится дополнительная переменная из вспомогательной задачи. Тогда рассчитываются оценки

$$æ_{pj}=\sum_{i=1}^{m} s_{pi}\, a_{ij}\ ,$$

где s_{pi} - элементы p-ой строки обратной матрицы базиса S^{-1};

a_{ij} - элементы j-го вектора условий матрицы A; $j \in J_q$.

2). Если $æ_{pj} \neq 0$, при некотором j = k, то вектор базиса p меняется на вектор условий k и происходит преобразование S^{-1}.

Если $æ_{pj} = 0$, для $\forall j \in J_q$, то, следовательно, имеет место ситуация б), т.е. rang A < m, и p-е ограничение можно исключить из матрицы условий.

И так далее подобная процедура выполняется для всех дополнительных переменных, находящихся в оптимальном базисе вспомогательной задачи. В дальнейшем целесообразно решать основную задачу с матрицей условий A размерности $r \times n$, где r = rang A < m.

М-метод построения исходного опорного решения

Способ одновременного решения задачи построения исходного опорного решения и основной задачи известен как М-метод решения задачи линейного программирования [87]. В этом случае исходная задача погружается в специальную задачу (М-задачу) вида

$$\begin{aligned}
&c^{T}x - M^{T} x_d \rightarrow \max,\\
&A\,x + E\,x_d = b,\\
&\;\; x \geq 0,\ \ x_d \geq 0.
\end{aligned}$$

Здесь M - достаточно большие положительные числа, которые имеют смысл "штрафов" на введенные дополнительные переменные x_d . Данная задача имеет очевидное исходное опорное решение $x_d = b$.

В процессе решения М-задачи симплекс-методом может быть установлена ее неразрешимость - целевая функция неограниченно возрастает на множестве допустимых решений, либо найдено оптимальное решение. Если в оптимальное решение входят положительные дополнительные переменные, то исходная задача неразрешима, т.е. нет допустимых решений, в противном случае (x_d = 0) оптимальное решение М-задачи является оптимальным решением исходной задачи.

Отметим, что для решения М-задачи линейного программирования нет необходимости задавать величину М заранее. В процессе решения при вычислении оценок $\delta_j = \pi^T a_j - c_j$, j = 1,...,n, следует полагать М больше любого сравниваемого с ним числа. Такая модификация симплекс-метода часто называется М-методом решения задач линейного программирования.

3.7. Вырожденность в задачах линейного программирования

Опорное решение, в котором одна или несколько базисных компонент равны 0, называется вырожденным.

Если задача вырожденная, то нарушается взаимно однозначное соответствие между опорными решениями и базисами, а, следовательно, и вершинами многогранника решений. Одному и тому же вырожденному опорному решению может соответствовать несколько различных базисов. Например, если опорное решение (x) имеет (p < m) отличных от 0 составляющих и, соответственно (m-p) равных 0 составляющих, то его базис состоит из (p) векторов условий, отвечающих ненулевым компонентам (x) и (m-p) векторов, соответствующих нулевым компонентам. Тогда вектор условий, соответствующий нулевой компоненте, можно заменить на любой небазисный вектор, и единственным требованием при этом будет линейная независимость векторов вновь образованного базиса.

Таким образом, одному и тому же вырожденному решению может соответствовать множество базисов (или несколько вершин многогранника). Данная ситуация может привести к зацикливанию алгоритма поиска оптимального решения (периодическому возвращению через несколько итераций к одному и тому же базису). Переход от одного вырожденного решения к другому происходит без изменения значения целевой функции, т.к. $\theta_r = 0$. Хотя, как отмечается в литературе, явление зацикливания достаточно редко, существует возможность модификации алгоритма с тем, чтобы гарантированно избежать такого зацикливания. При этом выбор вектора исключаемого из базиса (а именно на этом шаге и возникает неоднозначность) производится особым образом.

Рассмотрим один из способов выбора вектора, выводимого из базиса [87].

1. $t = 0$. Определяется множество $E^t = E^o$ индексов r, на которых достигается θ^o_r, где θ^o_r определяется в соответствии с (3.15)

$$\theta^o_r = \min_{æ_{ki}>0} \frac{x_{si}}{æ_{ki}}, \qquad i = 1,..,m.$$

2. Если E^t состоит из одного элемента r, то вектор a_k вводится на r-ю позицию базиса. Производится преобразование основной части таблицы 1.

В противном случае, если $|E^t| > 1$, то $t = t+1$; составляется множество E^t индексов r, на которых достигается θ^t_r, где θ^t_r вычисляется как

$$\theta^t_r = \min_{i \in E^{t-1}} \frac{s_{it}}{æ_{ki}}$$

здесь s_{it} - элементы t-го столбца матрицы S^{-1} из основной таблицы 1.

3. Производится переход на шаг 2.

Процесс повторяется до тех пор, пока не будет построено множество E^t, для которого θ^t_r достигается на единственной позиции.

Контрольные вопросы

1. Каким образом формализовано решение в задаче линейного программирования ?
2. В чем связь прямой и двойственной задач линейного программирования ? В чем смысл двойственных переменных ?
3. В чем отличие стандартной и канонической форм записи задач линейного программирования. В чем отличия двойственных к ним задач ?
4. Каким образом свести произвольную задачу линейного программирования к канонической форме (неравенства различных видов к равенствам; минимум к максимуму) ?
5. Как решить задачу линейного программирования, если переменные по условиям задачи могут принимать отрицательные значения ?

6. Что представляет множество допустимых решений задачи линейного программирования в n-мерном евклидовом пространстве? Почему ?

7. Где находится оптимальное решение задачи линейного программирования ? Почему ?

8. Как называется решение, соответствующее вершине многогранника решений? Чем оно отличается от других?

9. Что такое базис? Почему небазисные переменные равны 0?

10. Какие ситуации могут складываться в процессе решения задач линейного программирования ?

11. Каким образом в алгоритме симплекс-метода учитывается особенность представления опорного решения ?

12. Из каких шагов состоит отдельная итерация симплекс-метода ?

13. Что является признаком оптимальности опорного решения? Почему ?

14. Какой вектор условий целесообразно ввести в базис для улучшения текущего решения ?

15. Какой вектор целесообразно вывести из базиса для улучшения текущего решения ? Почему ? Как изменится значение целевой функции ?

16. Каким образом построить исходное опорное решение в различных задачах линейного программирования?

17. Каким образом при решении задач линейного программирования убедиться, что допустимых решений не существует. Почему такие ситуации могут быть ?

18. Каким образом выявить факт неограниченного возрастания целевой функции на множестве допустимых решений. В каких ситуациях это может быть ?

19. Что такое вырожденность в задачах линейного программирования ? К чему она может привести, почему ?

4. НЕЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

4.1. Особенности задач нелинейного программирования

Задачу математического программирования (2.3) представим в следующем виде

$$f(x) \to \max, \tag{4.1}$$

$$\varphi_i(x) \le b_i,\ \ i = 1,..., m, \tag{4.2}$$

$$x = (x_1,x_2,...,x_n)^т \ge 0. \tag{4.3}$$

Здесь выражение (4.1) задает целевую функцию, сопоставляющую каждому решению x число, характеризующее некоторое качество этого решения, а функциональные ограничения (4.2), (4.3) задают множество допустимых решений Δ. В отличие от задач линейного программирования, в которых целевая функция и функции, описывающие множество допустимых решений, имеют линейный вид, в задачах нелинейного программирования одна или несколько указанных функций нелинейные. В связи с этим возникает ряд особенностей задач нелинейного программирования по сравнению с линейными задачами.

1. По сравнению с рассмотренными ранее задачами линейного программирования, в которых оптимальное решение соответствует некоторым крайним точкам (вершинам) выпуклого многогранника, в задачах нелинейного программирования оптимальное решение может соответствовать произвольной точке области Δ. В этих задачах решение может находиться как в точке, расположенной на границе допустимой области, так и в точке, расположенной внутри области, что определяется свойствами целевой и ограничивающих функций.

2. Многосвязность допустимой области и многоэкстремальность задачи. В задачах нелинейного программирования при произвольных ограничениях (4.2) множество допустимых решений Δ может быть многосвязным, т.е. представлять собой объединение непересекающихся подмножеств. Решение в каждой такой области следует искать независимо от других, и выбирать из них наилучшее, однако установление факта многосвязности Δ, как правило, достаточно трудоемко.

Нахождение оптимума произвольной функции (4.1) в допустимой области Δ, даже если она задается линейными ограничениями, равно как и нахождение оптимума даже линейной функции на невыпуклом множестве Δ, может привести к тому, что в исходной оптимизационной задаче будет существовать несколько локальных экстремумов, т.е. другими словами такая задача будет многоэкстремальна (см.2.1). Существуют значительные трудности в определении локальных экстремумов, число которых в общем случае может быть достаточно велико.

3. Так как целевая функция и множество допустимых альтернатив в задачах нелинейного программирования описываются произвольными нелинейными функциями, в ряде приложений эти функции могут быть разрывными или недифференцируемыми. Это создает дополнительные трудности в решении таких задач и требует разработки специфических методов нахождения оптимальных решений.

В соответствии с рассмотренными особенностями задач нелинейного программирования, принято считать, что модель (задача) принятия решения является теоретически разработанной, если сформулированы необходимые и достаточные условия существования оптимального решения, и построены алгоритмы поиска решения, обладающие доказательствами сходимости. В этом смысле наиболее изученными и алгоритмически разработанными являются модели (задачи) выпуклого программирования, то есть задачи, в которых множество допустимых решений выпукло, а целевая функция вогнута (выпукла вверх). В этих условиях, как это было показано ранее (см. 2.3), оптимизационная задача является одноэкстремальной. Далее, рассматривая алгоритмы решения задач нелинейного программирования, в первую очередь будем иметь в виду именно задачи выпуклого программирования.

4.2. Алгоритмы решения задач нелинейного программирования

Эффективное решение различных задач нелинейного программирования может быть осуществлено на основе учета конкретных особенностей этих задач. При этом под эффективностью того или иного алгоритма, как правило, понимается объем вычислений, необходимый для получения приемлемого решения, который складывается из трудоемкости отдельной итерации и общего количества итераций вычислительного процесса. В настоящее время для решения различных задач нелинейного программирования разработано большое количество алгоритмов, построенных на использовании тех или иных методов, обладающих различными характеристиками и областью эффективной применимости. Общей особенностью указанных алгоритмов является то, что все они, как правило, для поиска оптимального решения в той или иной степени используют следующую итерационную схему:

$$x^{k+1} = x^k + h_k\, r(x^k),\quad k = 0,1,.... \tag{4.4}$$

Здесь

k - номер итерации;

x^k - решение на k-ой итерации;

$r(x^k)$ - направление, в котором изменяется значение x^k;

h_k - переменная, показывающая величину изменения (длину шага) вектора x^k в направлении $r(x^k)$; $h_k \geq 0$.

В алгоритмах подобного типа [13] необходимо анализировать (доказывать) сходимость (и скорость сходимости) последовательности $\{x^k, k=0,1,...\}$ к оптимальному решению x^*. Критерием окончания итерационного процесса служит выполнение неравенства

$$\| x^k - x^* \| \leq \varepsilon,$$

Поскольку значение x^*, как правило, заранее не известно, то часто используется критерий

$$\| x^{k+1} - x^k \| \leq \varepsilon, \text{ или } \| r(x^k) \| \leq \varepsilon, \text{ или } h_k \leq \varepsilon,$$

Таким образом, для реализации итерационного процесса (4.4) необходимо:

а) - задавать начальное значение x^o;

б) - на каждом k-ом шаге уметь вычислять направление $r(x^k)$, и величину шага h_k;

в) - задавать критерий окончания итерационного процесса ε.

С точки зрения области применимости алгоритмов различают универсальные (широкоспециализированные) и специализированные (узкоспециализированные) алгоритмы.

Универсальные алгоритмы предназначены для решения широкого класса задач нелинейного программирования. При построении этих алгоритмов используются самые общие свойства задач; выпуклость функций, дифференцируемость функций, замкнутость множества допустимых альтернатив и т. п.

Специализированные алгоритмы в той или иной степени учитывают специфику функций, описывающих множество допустимых альтернатив и целевой функции. Прежде всего, такая специфика выражается в линейности этих функций. Так, например, если ограничения задают выпуклый многогранник (описываются линейными уравнениями), т.е. имеют вид: $A\,x \le b$, $x \ge 0$, то в зависимости от вида целевой функции различают задачи:

- квадратичного программирования; целевая функция имеет вид: $f(x) = c^T x + x^T D x$, где D - неположительно определенная квадратная матрица;
- билинейного программирования; целевая функция имеет вид: $f(x) = c^T x + d^T y + x^T G\, y$;
- дробно-линейного программирования; целевая функция имеет вид:

$$f(x) = \frac{c^T x + a}{d^T x + b}$$

Учет отмеченных особенностей решаемых задач позволяет существенно повысить эффективность специализированных алгоритмов.

С точки зрения особенностей математических методов оптимизации, используемых при построении алгоритмов, можно различать [42]:

1) алгоритмы, использующие методы приведения исходной задачи нелинейного программирования с ограничениями к задаче оптимизации некоторой другой, отличной от целевой, функции в условиях отсутствия ограничений;

2) алгоритмы прямого решения исходной задачи;

3) специальные алгоритмы, использующие методы преобразования функций, процедуры случайного поиска и другие.

В дальнейшем рассмотрим некоторые методы и алгоритмы первых двух групп.

4.3. Методы приведенной безусловной оптимизации

Методы данной группы строятся на основе использования концепции "погружения", когда исходная задача нелинейного программирования (4.1)-(4.3) с ограничениями приводится к задаче оптимизации некоторой функции F(x) в условиях отсутствия ограничений, и далее используются методы безусловной оптимизации. В качестве таких функций F(x) наиболее часто используются функции Лагранжа и штрафные функции.

4.3.1. Методы безусловной оптимизации

Представим задачу безусловной оптимизации в виде

$$x^* = \arg \max_{x \in R^n} F(x) . \qquad (4.5)$$

В основе алгоритмов решения задачи (4.5) лежит итеративный процесс (4.4):

$$x^{k+1} = x^k + h_k\, r(x^k),\ k = 0,1,....$$

Здесь k - номер итерации; x^k - решение на k-ой итерации; $r(x^k)$ - направление, в котором изменяется значение x^k; h_k - величина изменения (шаг) вектора x^k в направлении $r(x^k)$; $h_k \geq 0$.

В зависимости от способа вычисления направления $r(x^k)$, способа вычисления шага h_k строятся различные алгоритмы решения задачи безусловной оптимизации.

<u>Градиентный метод</u>

В качестве $r(x^k)$ используется направление, в котором наиболее сильно возрастает целевая функция. Это направление задается градиентом функции $\nabla F(x^k)$. Суть метода состоит в последовательном приближении к оптимальному решению x^* на основе проведения ряда итераций вида

$$x^{k+1} = x^k + h_k\, \nabla F(x^k),\ \ k = 0,1,.... \qquad (4.6)$$

Алгоритмы, построенные на основе градиентного метода, носят итерационный характер; итерация состоит в выполнении ряда шагов.

0. Исходное состояние. Задается начальная точка $x^o \in R^n$, некоторое малое $\varepsilon \geq 0$; $k = 0$.

1. Рассчитывается градиент в точке x^k; $\nabla F(x^k)$.

2. Задается шаг h_k.

3. Находится $x^{k+1} = x^k + h_k \nabla F(x^k)$.

4. Проверяется условие $\|x^{k+1} - x^k\| \le \varepsilon$, или $\|\nabla F(x^k)\| \le \varepsilon$.
Если данное условие выполняется, то счет окончен; если нет, то $k = k+1$ и производится переход на шаг 2.

В пункте 2 задается шаг, обеспечивающий смещение решения в направлении градиента (возрастания целевой функции). Шаг h_k следует выбирать таким, чтобы обеспечивалось приращение целевой функции в направлении $\nabla F(x^k)$, т.е. должно выполняться условие $F(x^{k+1}) > F(x^k)$. Вместе с тем, если h_k очень велик, то возможны ситуации, когда $F(x^{k+1}) < F(x^k)$; если h_k очень мал, то процесс, описываемый (4.7), сходится достаточно долго (см. рис.4.1).

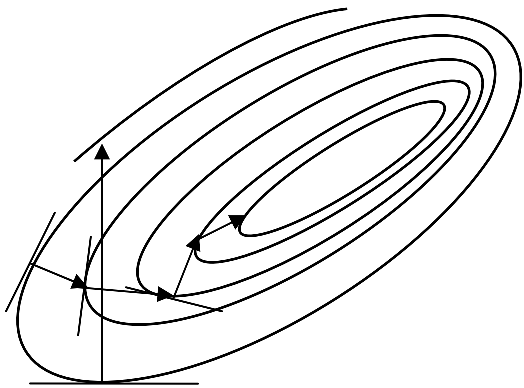


Рис. 4.1.

В зависимости от способа выбора h_k на каждой итерации выделяют различные модификации алгоритма.

1. Постоянный шаг.

Задается $h_k = h = const$, при этом должно выполняться условие

$$F(x^{k+1}) = F(x^k + h_k\nabla F(x^k)) > F(x^k).$$

Пусть F(x) дифференцируема в окрестности точки x^k (см. 2.2). Тогда

$$F(x) = F(x^k) + (\nabla F(x^k),(x - x^k)) + o(\|x - x^k\|),$$

откуда

$$F(x^{k+1}) - F(x^k) \ge \nabla^T F(x^k)\,(x^{k+1} - x^k),$$

или, подставляя в последнее выражение значение x^{k+1} из (4.6), получаем

$$F(x^{k+1}) - F(x^k) \geq h_k \| \nabla F(x^k) \|^2. \quad (4.7)$$

Тогда, если известно оптимальное значение $F^* = \max F(x)$, или $F^* = \sup F(x)$, то шаг можно выбирать из условия

$$F^* - F(x^k) = h_k \| \nabla F(x^k) \|^2.$$

Если по мере продвижения к оптимуму условие (4.7) перестает выполняться, то производится дробление шага.

2. Наискорейший подъем.

Если подставить в выражение для F(x) значение $x=x^{k+1}$ в соответствии с (4.6), то получим выражение $F(x^k+h_k\nabla F(x^k))$, как функцию от величины шага. Следовательно, можно выбирать величину шага h_k на каждой итерации, исходя из условия максимизации функции, т.е.

$$h_k = \max_{h\geq 0} F(x^k+h\nabla F(x^k)) \quad (4.8)$$

Последнее выражение представляет собой задачу одномерной оптимизации.

<u>Метод покоординатной (покомпонентной) оптимизации</u>

Сущность метода [13] заключается в том, что в качестве направления $r(x^k)$ последовательно выбираются направления изменения координат (компонент) вектора x. И, если целесообразность увеличения (уменьшения) i-ой компоненты x существует, то такое увеличение (уменьшение) происходит на величину h_k. Таким образом, в качестве направления на k-м шаге выбирается некоторый единичный вектор $e_i(k)$, все компоненты которого, кроме i-ой равны 0 (i-я компонента вектора $e_i(k)$ равна 1). Если в некотором цикле, в котором проверяется целесообразность изменения всех компонент x_i, i = 1,...,n, вектор x не изменился, то происходит дробление шага $h_{k+1} = \alpha h_k$.

Тогда алгоритм покоординатной оптимизации состоит из следующих шагов.

0). Исходное состояние. Задается некоторая точка $x^o \in R^n$. Задаются числа $h_o>0$; $0<\alpha<1$; $\delta>0$; $\varepsilon>0$. Номер цикла c = 0; номер итерации k = 0.

1). Вычисляется номер проверяемой координаты (компоненты) вектора x на итерации k.

$$i(k) = k - cn + 1.$$

2). Вычисляется значение функции $F(x^k + h_k e_i(k))$, если $F(x^k + h_k e_i(k)) > F(x^k)$, то $x^{k+1} = x^k + h_k e_i(k)$.

В противном случае вычисляется $F(x^k - h_k e_i(k))$, если $F(x^k - h_k e_i(k)) > F(x^k)$, то $x^{k+1} = x^k - h_k e_i(k)$.

Если ни одним из способов значение целевой функции улучшить не удается, то итерация k считается неудачной и $x^{k+1} = x^k$, в противном случае итерация считается удачной.

3). $k = k + 1$; вычисляется $c' = [k/n]$, где квадратные скобки имеют смысл взятия целой части отношения k/n.

Если $c' = c$, т.е. номер цикла не изменился, то производится переход на шаг 1.

Если $c' > c$, то $c = c'$. Оценивается удачность цикла; цикл считается удачным, если в нем была удачная итерация.

Если цикл удачный, то производится переход на шаг 4; если цикл неудачный, то производится переход на шаг 5.

4). Проверяется критерий окончания вычислительного процесса. Если выполняется условие $\|x^{k+1} - x^k\| \le \varepsilon$, то работа алгоритма заканчивается, если данное условие не выполняется, то производится переход на шаг 1.

5). Производится дробление шага $h_{k+1} = \alpha h_k$, где $0<\alpha<1$. Если $h_{k+1} \le \delta$, то вычислительный процесс заканчивается; если $h_{k+1} > \delta$, то производится переход на шаг 1.

4.3.2. Метод функции Лагранжа

Исторически первым способом сведения задачи с ограничениями к задаче безусловной оптимизации явилось использование функции Лагранжа $L(x,\mu)$

$$L(x, \mu) = f(x) + \mu^T(b - \varphi(x)) = f(x) + \sum_{i=1}^{m} \mu_i (b_i - \varphi_i(x)),$$

где $\mu_i \ge 0$, $i=1,...,m$ - множители Лагранжа.

В соответствии с теоремой Куна-Таккера необходимым и достаточным условием оптимальности является то, что точка (x^*,μ^*) - седловая точка функции Лагранжа, т.е. удовлетворяет условию

$$\max_x \min_\mu L(x, \mu) = L(x^*, \mu^*) = \min_\mu \max_x L(x, \mu)$$

При фиксированном μ левая часть представляет собой прямую задачу математического программирования, при фиксированном x правая часть представляет собой двойственную задачу. На этой основе можно строить итеративный процесс, когда на каждом k-м шаге при фиксированных x^k и μ^k решается пара двойственных задач вида

$$x^{k+1} = \arg\max_{x\geq 0} L(x, \mu^k) \quad \text{и} \quad \mu^{k+1} = \arg\min_{\mu\geq 0} L(x^k, \mu)$$

Эти задачи являются задачами безусловной оптимизации и могут решаться, например, градиентными методами. Тогда

$$x^{k+1} = (x^k + A_k \nabla_x L(x^k,\mu^k))^+,$$
$$\mu^{k+1} = (\mu^k - B_k \nabla_\mu L(x^k,\mu^k))^+,$$

здесь A_k, B_k - шаговые множители; операция $(^+)$ позволяет обеспечить выполнение условий $x\geq 0$, $\mu\geq 0$ и имеет следующий смысл

$$x_i^{k+1} = \max\{0, x_i^k + A_k \nabla_x L(x_i^k,\mu^k)\},$$
$$\mu_i^{k+1} = \max\{0, \mu_i^k - B_k \nabla_\mu L(x^k,\mu_i^k)).$$

Метод характеризуется достаточно медленной сходимостью, однако, существуют способы увеличения скорости сходимости к оптимальному решению за счет специального выбора значений шаговых множителей A_k, B_k.

4.3.3. Штрафные функции

Исходная задача условной оптимизации сводится к последовательности задач безусловной оптимизации функций [20]

$$F_k(x, \mu) = f(x) - S_k(x, \mu_k), \quad k = 1,2,3,....$$

Здесь $S_k(x,\mu_k)$, $k = 1,2,3,...$ - штрафные функции, задаваемые таким образом, чтобы обеспечить выполнение условия $x\in\Delta$. Тогда $F_k(x,\mu)$ мало отличаются от $f(x)$, если $x\in\Delta$, и быстро убывают с ростом k, если $x \in R^n \setminus \Delta$. Величина штрафа определяется подбором коэффициентов μ_k.

Различают три основных способа формирования штрафных функций:

- внешние штрафные функции;
- внутренние штрафные (барьерные) функции;
- комбинированный метод.

Внешние штрафные функции определены на всем пространстве R^n, причем в допустимой области они равны 0, а вне ее принимают положительные значения и быстро возрастают с ростом k. Например, в качестве внешней штрафной функции может использоваться функция, характеризующая взвешенную сумму квадратов невязок

$$S_k(x, \mu_k) = \sum_{i=1}^{m} \mu_{ki} (\max \{0, (\varphi_i(x) - b_i)\})^2.$$

Здесь, придавая коэффициентам μ_k, k=1,2,.... различные значения, можно повысить скорость сходимости метода. Широкий класс штрафных функций задается выражением

$$S_k(x, \mu_k) = -1 + \exp \left(\sum_{i=1}^{m} \mu_{ki} \max \{0, (\varphi_i(x) - b_i)\}\right).$$

Очевидно, что штрафных функций может быть достаточно много, однако, для повышения скорости сходимости вид штрафных функций должен быть согласован с видом функций $\varphi_i(x)$, i = 1,...,m и видом функции f(x).

Внутренние штрафные (барьерные) функции используются тогда, когда поиск оптимального решения необходимо вести, не выходя за пределы допустимой области Δ. В этой связи для конструирования барьерных функций необходимо выбирать такие функции $S_k(x,\mu_k)$, которые принимают значение 0, если $x\in\Delta$, и неограниченно возрастают при приближении x к границе Δ. В качестве барьерных функций может использоваться, например, обратная функция барьера

$$S_k(x, \mu_k) = \sum_{i=1}^{m} \mu_{ki} (b_i - \varphi_i(x))^{-1}.$$

или логарифмическая функция барьера

$$S_k(x, \mu_k) = -\sum_{i=1}^{m} \ln \min \{1, \mu_{ki} (b_i - \varphi_i(x))\}.$$

Характерной особенностью метода барьерных функций является необходимость выбора начального приближения x, такого чтобы $x\in\Delta$. Существенные трудности возникают в ситуациях, когда в описание допустимой области Δ входят ограничения типа равенств. Тем не менее, метод барьерных функций, наряду с методом внешних штрафных функций достаточно широко используется при поиске оптимальных решений.

4.4. Методы прямой условной оптимизации

Методы прямой условной оптимизации предназначены для непосредственного решения задачи выпуклого программирования в условиях ограничений, описывающих множество допустимых решений Δ.

Итак, пусть решается задача нелинейного программирования

$$f(x) \to \max_{x \in \Delta}, \tag{4.9}$$

где Δ определяется как

$$\varphi_i(x) \le b_i, \;\; i = 1,..., m, \tag{4.10}$$

$$x \ge 0. \tag{4.11}$$

Использование прямых методов условной оптимизации строится на основе использования итеративной схемы (4.4)

$$x^{k+1} = x^k + h_k\, r(x^k), \;\; k = 0,1,....$$

Однако, при построении методов данной группы на каждом k-м шаге итеративного процесса выбор рационального направления $r(x^k)$ и величины шага h_k производится таким образом, чтобы обеспечивалась допустимость очередного решения, т.е. $x^{k+1} \in \Delta$.

4.4.1. Метод условного градиента

Существо метода условного градиента [13] состоит в том, что, если известна некоторая точка $x^k \in \Delta$, то направление возрастания целевой функции может задаваться некоторой внутренней или крайней точкой $x \in \Delta$, которая ищется в области Δ в направлении, задаваемом градиентом $\nabla f(x)$. Действительно, пусть h_k=1,тогда

$$x = x^k + r(x^k), \text{ или отсюда } \; r(x^k) = x - x^k.$$

С другой стороны из определения (2.2) дифференцируемости функции в точке x^k

$$f(x) = f(x^k) + \nabla^т f(x^k)(x - x^k) + o(\|x - x^k\|)$$

или

$$f(x) - f(x^k) > \nabla^т f(x^k)(x - x^k).$$

Отсюда приращение целевой функции $f(x) - f(x^k)$ при переходе к следующей точке x можно максимизировать на основе специального выбора точки $x = x^k_*$, производимого в результате решения следующей задачи

$$x^k_* = \arg \max_{x \in \Delta} \nabla^T f(x^k)(x - x^k), \tag{4.12}$$

Таким образом, x^k_* задает направление максимального приращения целевой функции из точки x^k.

Если x^k_* - граничная точка Δ, то следующую точку, точку x^{k+1} целесообразно искать на отрезке, соединяющем x^k и x^k_*, т.е.

$$x^{k+1} = h_k x^k_* + (1-h_k) x^k, \quad \text{где} \quad 0 \le h_k \le 1,$$

или

$$x^{k+1} = x^k + h_k (x^k_* - x^k) = x^k + h_k r(x^k), \quad \text{где } 0 \le h_k \le 1.$$

Величину шага h_k следует выбирать из условия максимизации значения целевой функции в точке x^{k+1}, т.е.

$$h_k = \max_{0 \le h \le 1} f(x^k + h(x^k_* - x^k)) \tag{4.13}$$

Последняя задача представляет собой задачу одномерной оптимизации.

Основные трудности использования метода условного градиента при решении задач нелинейного программирования состоят в решении задачи (4.12), и, в обще говоря, в ряде случаев задача оптимизации функции градиента на Δ может быть сравнима по трудоемкости с исходной задачей. Однако для некоторых классов задач, в частности, для задач квадратичного программирования, метод условного градиента дает существенный выигрыш.

<u>Задача квадратичного программирования</u>

В случае, если рассматривается задача квадратичного программирования, то (4.9)-(4.11) можно переписать в виде

$$f(x) = c^T x + x^T D x \to \max, \tag{4.14}$$
$$A x \le b, \tag{4.15}$$
$$x \ge 0, \tag{4.16}$$

где D-симметрическая отрицательно полуопределенная матрица.

В этом случае задача поиска наилучшего направления (4.12) является задачей линейного программирования. Рассмотрим алгоритм решения задачи (4.14)-(4.16) по методу условного градиента.

0). Исходное состояние. Задается $x^o \in \Delta$, $\varepsilon > 0$; $k = 0$.

1). Рассчитывается $\nabla f(x^k)$.

2). Ищется x^k_* - направление максимального приращения целевой функции из точки x^k. Для этого решается задача линейного программирования вида

$$x^k_* = \arg \max_{x \in \Delta} \nabla^т f(x^k)(x - x^k) = \arg \max_{x \in \Delta} \nabla^т f(x^k)\, x\ ,$$

Здесь Δ определяется ограничениями (4.15),(4.16).

3). Решается задача одномерной оптимизации (4.13)

$$h_k = \max_{0 \le h \le 1} f(x^k + h(x^k_* - x^k))$$

4). Вычисляется значение x^{k+1}

$$x^{k+1} = h_k\, x^k_* + (1 - h_k)\, x^k = x^k + h_k\, (x^k_* - x^k).$$

5). Производится оценка $\|x^{k+1} - x^k\| \le \varepsilon$, если данное условие выполняется, то производится окончание работы алгоритма, если нет, то $k = k+1$ и производится переход на шаг 1.

4.4.2. Метод возможных направлений

(Зойтендейк Г., 1960)

Основное содержание метода [36] заключается в том, что на каждом шаге итерационного процесса $x^{k+1} = x^k + h_k\, r(x^k)$ выбирается направление $r(x^k)$, позволяющее повысить значение целевой функции (4.9) и подходящее с точки зрения удовлетворения ограничений (4.10),(4.11). В выбранном направлении совершается шаг h_k, величина которого такова, что позволяет максимально повысить значение целевой функции и не выводит за границы допустимой области.

В о з м о ж н ы м направлением возрастания целевой функции в точке x^k будем называть направление $r(x^k) = (x - x^k)$, такое, что 1) $x^{k+1} = x^k + h_k(x - x^k) \in \Delta$,

2) $f(x^{k+1}) = f(x^k + h_k(x - x^k)) > f(x^k)$

при $\forall h_k \in (0, h^м_k]$, где $h^м_k > 0$.

<u>Выбор возможного направления</u>

Обозначим возможное направление через γ^k, т.е. $\gamma^k = r(x^k) = (x - x^k)$. Тогда из условия (2.2) дифференцируемости целевой функции следует, что

$$f(x) = f(x^k) + \nabla^т f(x^k)\, \gamma^k + o(\|x - x^k\|)$$

или $f(x) - f(x^k) > \nabla^т f(x^k)\,\gamma^k$.

В этом случае γ^k можно найти в результате решения задачи

$$\gamma^k = \arg\max_{\gamma} \nabla^т f(x^k)\,\gamma$$

При этом точка $x = \gamma^k + x^k$ должна быть допустима, т.е. должна удовлетворять условиям (4.10), (4.11).

Из дифференцируемости функции $\varphi_i(x)$, описывающей множество допустимых решений, i-е условие в (4.10), можно переписать в виде

$$\varphi_i(x) = \varphi_i(x^k) + \nabla^т\varphi_i(x^k)\,\gamma + o(\|x - x^k\|) \le b_i,\ i=1,...,m,$$

или

$$\varphi_i(x^k) + \nabla^т\varphi_i(x^k)\,\gamma < b_i,\ \ i = 1,..., m. \qquad (4.17)$$

Если x^k - внутренняя точка, то в направлении градиента (максимальном направлении возрастания целевой функции) можно сделать некоторый шаг (пусть даже очень маленький). Тогда при поиске наилучшего направления γ следует учитывать только такие ограничения вида (4.17), которые выполняются в точке x^k как строгие равенства ($\varphi_i(x^k) = b_i$). Следовательно, (4.17) можно переписать

$$\nabla^т\varphi_i(x^k)\,\gamma < 0,\ \ i \in I_k = \{r|\ \varphi_r(x^k) = b_r\}. \qquad (4.18)$$

Введем переменную $z \le \nabla^т f(x^k)\gamma$. Тогда задачу выбора наилучшего направления можно представить в виде

$$z \to \max, \qquad (4.19)$$
$$-\nabla^т f(x^k)\,\gamma + z \le 0, \qquad (4.20)$$
$$\nabla^т\varphi_i(x^k)\,\gamma + z \le 0,\ i \in I_k = \{r|\ \varphi_r(x^k) = b_r\} \qquad (4.21)$$
$$-1 \le \gamma_j \le 1,\ \ j = 1,\dots,n. \qquad (4.22)$$

Задача (4.19)-(4.22) представляет собой задачу линейного программирования с вектором неизвестных переменных $(\gamma^т, z)^т$. В результате решения данной задачи будет выбираться направление γ, максимально близкое к направлению градиента с учетом активных ограничений (4.21).

Так как решение $(0^т,\ 0)^т$ является допустимым решением задачи, то, очевидно, что целевая функция ограничена снизу и, следовательно, $z \ge 0$.

Если в результате решения задачи (4.19)-(4.22) получаем значение $z > 0$, то это означает, что можно повысить значение целевой функции в направлении, задаваемом γ, тогда $\gamma^k = \gamma$.

Если $z = 0$, следовательно, не существует направления, позволяющего повысить значение целевой функции, тогда оптимальное решение $x^* = x^k$.

Если x^k внутренняя точка, то ограничения (4.21) отсутствуют, и тогда направление γ в соответствии с (4.20) определяется как

$$\gamma = \nabla f(x^k) \times \frac{z}{\left\|\nabla f\left(x^k\right)\right\|^2},$$

т.е. совпадает с направлением градиента.

<u>Задание величины шага</u>

Итак, вычислено наилучшее направление γ^k, в котором следует изменять значения x^k для увеличения целевой функции. В этом направлении можно сделать шаг h_k, но не более некоторой максимальной величины $h^м_k$, которая определяется границами допустимой области Δ. Тогда $h^м_k$ можно вычислить как

$$h^м_k = \arg\max_h \{ h \mid \varphi_i (x^k + h\,\gamma^k) \le b_i,\ i = 1,\ldots, m\}.$$

Далее шаг h_k выбирается по аналогии с методом наискорейшего спуска, как

$$h_k = \arg\max_{0 \le h \le h^м_k} f(x^k + h\,\gamma^k).$$

Последнее выражение представляет собой задачу одномерной оптимизации, которая может быть решена соответствующими методами.

Контрольные вопросы

1. В чем состоят особенности задач нелинейного программирования по сравнению с задачами линейными ?
2. Каким образом решить задачу с ограничениями на основе решения задачи без ограничений ?
3. В чем суть итерационной схемы решения задач нелинейного программирования. Как она используется в алгоритмах безусловной оптимизации ?
4. Каким образом выбирается направление увеличения целевой функции в алгоритмах прямой условной оптимизации ?
5. Какими способами может выбираться величина шага ?

5. ДИСКРЕТНОЕ ПРОГРАММИРОВАНИЕ

5.1. Постановка задач дискретного программирования

Задачей дискретного программирования [50] называется задача нахождения экстремума функции, определенной на дискретном множестве, состоящем из изолированных точек. Основной класс задач дискретного программирования - задачи, в которых на переменные, являющиеся компонентами вектора, описывающего решение, наложено условие целочисленности. Теория и методы решения таких задач объединяются в рамках направления, которое называется ц е л о ч и с л е н н о е программирование и предназначено для решения задач математического программирования в виде (2.1), в которых накладываются дополнительные условия целочисленности на все или часть компонент вектора, описывающего решение. Тогда задача целочисленного программирования имеет вид

$$x^* = \arg \max_{x \in \Delta} f(x), \text{ где } \Delta = \{x \in Z^n \mid \varphi_i(x) \le b_i,\ i = 1, \ldots, m\}, \quad (5.1)$$

где Z^n - n-мерное пространство векторов с целочисленными компонентами.

Если только часть (s) компонент вектора x должна быть целочисленна, т.е. $\Delta \subset Z^s \times R^q$, то говорят о задачах ч а с т и ч н о целочисленного программирования.

Частным случаем задач целочисленного программирования являются задачи, в которых переменные принимают значения из множества {0,1}. Такие задачи называются задачами булевого (бивалентного, логического) программирования. Особое место задач булевого программирования определяется тем, что любую задачу целочисленного программирования, в которой значения переменных ограничены некоторым числом, можно свести к булевой задаче путем замены переменных. Действительно, пусть $x_j \in Z$, $x_j = 0,\dots,v_j$, тогда x_j можно заменить совокупностью переменных $x_{ij} \in \{0,1\}$. При этом

$$x_j = \sum_{i=0}^{v_j} x_{ij}, \quad \sum_{i=0}^{v_j} x_{ij} = 1, \quad x_{jj} \in \{0,1\}$$

Специфика задач целочисленного (дискретного) программирования заключается в том, что здесь происходит разрыв условий двойственности в каждой точке Δ, вследствие чего условия теоремы Куна-Таккера не выполняются, и для задачи (5.1) не существует двойственной задачи. В этой связи методы, развитые для задач линейного и нелинейного программирования, не применимы в общем случае для задач целочисленного программирования.

Наиболее изученным классом задач целочисленного программирования являются задачи целочисленного линейного программирования, обладающие рядом свойств, позволяющих строить эффективные алгоритмы решения

$$x^* = \arg \max_{x \in \Delta} f(x), \text{ где } \Delta = \{x \in Z^n_+ \mid \sum_{j=1}^{n} a_{ij}\, x_j \le b_i,\ i = 1,\dots, m\}.$$

Используемые в настоящее время методы решения задач целочисленного программирования, можно разбить на пять групп.

***1 группа методов*.** Используется традиционный математический аппарат решения непрерывных задач. Целочисленность достигается округлением результата до ближайшего целого из области допустимых решений. Несмотря на кажущуюся простоту метода, при округлении возникают трудности, связанные, прежде всего, с тем, что количество вариантов округления может быть достаточно велико. В результате округления необходимо получить допустимое решение, а это при большом числе компонент вектора, описывающего решение, довольно не тривиальная задача. Кроме того, произвольное округление компонент может привести к довольно значительным отклонениям от оптимального решения. Более того, при округлении можно не обнаружить допустимого решения, которое на самом деле существует.

2 группа методов. Выделяется класс задач, в которых использование методов решения непрерывных задач математического программирования автоматически приводит к получению целочисленных решений.

<u>3 группа методов</u>. Методы отсечения. Решается задача без ограничения на целочисленность, по результатам решения которой в исходную постановку вводятся дополнительные ограничения, отсекающие нецелочисленные решения. Процесс продолжается до получения оптимального целочисленного решения.

<u>4 группа методов</u>. Комбинаторно-эвристические методы. Все множество альтернатив разбивается на подмножества, и вводятся оценки, позволяющие отбрасывать подмножества, в которых не содержится оптимального решения. Процесс продолжается до выявления оптимума.

<u>5 группа методов</u>. Приближенные методы. Здесь объединяется большая группа методов, связанных с приближенной оптимизацией (неточное нахождение оптимума, приближенное удовлетворение целочисленности, и т.п.). Здесь же рассматриваются методы случайного поиска оптимального решения.

Далее рассмотрим более подробно методы второй, третьей и четвертой групп.

5.2. Целочисленные и почти целочисленные многогранники решений

Рассмотрим задачу целочисленного линейного программирования в виде

$$L = c^{T}x \rightarrow \max, \qquad (5.2)$$

$$A\,x \le b, \qquad (5.3)$$

$$x \ge 0, \qquad (5.4)$$

$$x = (x_1,\dots,x_j,\dots,x_n)^{T},\ \forall x_j \text{ - целое.} \qquad (5.5)$$

Как было показано в 3.2, оптимальное решение x^* задачи линейного программирования (5.2)-(5.4) соответствует вершине выпуклого многогранника M(A,b), задаваемого ограничениями (5.3), (5,4). Тогда, если все решения, соответствующие вершинам многогранника M(A,b) целочисленны (говорят, что вершины целочисленны, или многогранник - ц е л о ч и с л е н н ы й), то и оптимальное решение x^* будет целочисленным.

Пусть M(A,b) - выпуклый многогранник, определяемый (5.3), (5,4). Обозначим через G множество целочисленных решений, удовлетворяющих (5.3), (5.4), т.е. принадлежащих M(A,b). Тогда в ы п у к л о й о б о л о ч к о й G - M(G) называется пересечение всех выпуклых множеств, содержащих G.

Утверждение. Пусть G множество целочисленных точек, принадлежащих M(A,b). Тогда множество решений задачи (5.2)-(5.5) содержится в выпуклой оболочке M(G).

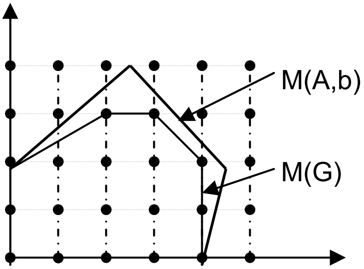


рис. 5.1

Очевидно, что M(G) $\subseteq$ M(A,b). Следовательно, если выделить класс задач целочисленного линейного программирования, для которых M(G) = M(A,b), то для нахождения оптимального решения в таких задачах можно использовать методы линейного программирования.

Матрица A называется а б с о л ю т н о у н и м о д у л я р н о й, если любой минор (любого порядка) матрицы A равен 0,1,-1.

Утверждение. Многогранник M(A,b), задаваемый ограничениями (5.3), (5.4), является целочисленным при любом целочисленном b тогда и только тогда, когда матрица A абсолютно унимодулярна [58].

В настоящее время не сформулировано конструктивных (легко проверяемых) необходимых и достаточных условий абсолютной унимодулярности матриц, тем не менее, определенные критерии, которым должны удовлетворять такие матрицы, можно задать. Так, в частности, из определения абсолютной унимодулярности следует, что элементы A должны принадлежать множеству {-1,0,1}. Сформулированы также достаточные условия абсолютной унимодулярности A [58].

Достаточные условия абсолютной унимодулярности.

Матрица A ={ a_{ij} } с элементами $a_{ij} \in \{-1,0,1\}$ является абсолютно унимодулярной, если

1) каждый столбец A содержит не более двух отличных от 0 элемента;

2) множество строк матрицы A можно разбить на два непересекающихся подмножества строк R_1 и R_2 такие, что

а) отличные от 0 элементы некоторого столбца A одного знака принадлежат разным подмножествам строк;

б) отличные от 0 элементы некоторого столбца A разного знака принадлежат одному из подмножеств строк.

Сформулированные достаточные условия абсолютной унимодулярности являются в известной степени строгими, тем не менее, для ряда моделей принятия решений они выполняются. Так для рассмотренных в 2.2 транспортной модели и модели назначений приведенные достаточные условия выполняются. Действительно, пусть в транспортной модели (см. 2.2) s=2, q=3, тогда матрица A условий такой задачи будет иметь вид

$$A = \begin{Vmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{Vmatrix}$$

В каждом столбце содержится 2 отличных от 0 элемента. Строки такой матрицы можно разбить на два подмножества $R_1 = \{1, 2\}$ и $R_2 = \{3,4,5\}$, которые удовлетворяют предъявляемым требованиям.

Многогранник M(A,b) называется *почти целочисленным*, если все целочисленные точки M(A,b) являются его вершинами, и множество целочисленных вершин связно [29].

Тогда для нахождения оптимального решения с использованием симплекс-метода на каждой итерации необходимо анализировать целочисленность вершины, в которую осуществляется переход. Если она целочисленна, то переход осуществляется; если нет, то ищется другая смежная целочисленная вершина, позволяющая повысить значение целевой функции. Процесс заканчивается, когда не найдется целочисленной вершины, позволяющей увеличить целевую функцию.

Утверждение. Многогранник $M(A,e)$, где e - вектор, все компоненты которого равны 1, а элементы матрицы A принимают значения 0 или 1, является почти целочисленным.

5.3. Методы отсечения в задачах дискретного программирования. Алгоритм Р.Гомори

Основная идея методов отсечения [42, 49, 50, 84] состоит в том, что решается задача без ограничения на целочисленность, по результатам анализа решения которой в исходную постановку вводятся дополнительные ограничения, отсекающие нецелочисленные и оставляющие целочисленные решения. Процесс продолжается до получения оптимального целочисленного решения (т.е. целочисленная точка становится крайней). Методы отсечения позволяют решать как задачи целочисленного, так и задачи частично целочисленного программирования; они применимы в тех случаях, когда требование целочисленности распространяется и на целевую функцию; данные методы позволяют получить решение за конечное число шагов.

Рассмотрим задачу целочисленного линейного программирования в виде

$$L = \sum_{j=1}^{n} c_j x_j \to \max \tag{5.6}$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i,\ i = 1,\ldots,m, \tag{5.7}$$

$$x_j \ge 0, \qquad j = 1,\ldots,n, \tag{5.8}$$

$$x_j \text{ - целое},\ j = 1,\ldots,n. \tag{5.9}$$

Методы отсечения связаны с систематическим введением дополнительных ограничений (отсечений), целью которого является сведение допустимой области, определяемой (5.7),(5.8), к выпуклой оболочке ее целочисленных точек в окрестности оптимального решения. Вводимое ограничение (отсечение) должно удовлетворять требованиям:

- ограничение должно отсекать нецелочисленную точку;
- ограничение должно оставлять в допустимой области все целочисленные точки;

- ограничение, соответствующее отсечению, должно быть линейным, чтобы задача оставалась в классе задач линейного программирования;
- ограничение проходит через целочисленную точку (хотя и не обязательно в допустимой области).

Первый алгоритм такого рода был предложен Р.Гомори. Рассмотрим основные идеи этого алгоритма [84].

Отсечение Р.Гомори

Решается задача (5.6)-(5.8). Находится решение x, соответствующее вершине M(A,b).

$x = (x_s, x_q)^T$, где x_s - базисные переменные;

x_q - небазисные переменные;

$x_s = \| x_{si} \|$, $i \in J_s$, J_s - номера базисных переменных;

$x_q = \| x_{qj} \|$, $j \in J_q$, J_q - номера небазисных переменных.

Матрицу условий A можно представить в виде $A = \| S\ Q \|$, тогда

$$S x_s + Q x_q = b \quad \text{или} \quad x_s + S^{-1} Q x_q = S^{-1} b,$$

откуда

$$x_s = S^{-1} b - S^{-1} Q x_q, \text{ причем } x_s \geq 0,\ x_q = 0.$$

Пусть i-я компонента x_s нецелочисленна. Скомпенсировать дробную часть x_s можно за счет придания компонентам x_q отличных от 0 значений. Обозначим

$$S^{-1} b = \beta,\ \beta = \| \beta_i \|,\ S^{-1} Q = \alpha,\ \alpha = \| \alpha_{ij} \|.$$

Тогда

$$x_{si} = \beta_i - \sum_{j \in j_q} \alpha_{ij} x_{qj}.$$

Итак, i-ая компонента $x_{si} = \beta_i$ - нецелочисленна. Представим β_i, а соответственно и α_{ij} в виде суммы целой и дробной части

$$\beta_i = [\beta_i] + \{\beta_i\};\ \alpha_{ij} = [\alpha_{ij}] + \{\alpha_{ij}\},$$

где $[\beta_i]$ - целая часть β_i, не превосходящая β_i (соответственно, $[\alpha_{ij}]$ - целая часть α_{ij}); $\{\beta_i\}$ - дробная часть β_i (соответственно $\{\alpha_{ij}\}$ - дробная часть α_{ij}); $\forall \{\beta_i\} > 0$, $\forall \{\alpha_{ij}\} > 0$.

Тогда

$$x_{si} = [\beta_i] + \{\beta_i\} - \sum_{j \in j_q} [\alpha_{ij}] x_{qj} - \sum_{j \in j_q} \{\alpha_{ij}\} x_{qj}.$$

Необходимо скомпенсировать дробную часть x_{si}, тогда некоторые (или все) x_{qj} будут целые и больше 0. Перепишем последнее уравнение в виде

$$x_{si} - [b_i] + \sum_{j \in j_q} [a_{ij}]\, x_{qj} = \{\beta_i\} - \sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. \qquad (5.10)$$

Здесь левая часть представляет собой целое число, соответственно правая часть должна быть целым числом. При этом возможны две ситуации

- первая: $\{\beta_i\} - \sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. > 0;$
- вторая: $\{\beta_i\} - \sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. \le 0$

Рассмотрим эти ситуации, начнем с первой. Так как $\{\beta_i\} > 0$ и $\{\alpha_{ij}\} > 0$, то эта ситуация возможна только, если левая часть менее 1, т.е. она является дробью, что невозможно, т.к. левая часть равенства (5.10) - целое число. Следовательно, имеет место вторая ситуация, т.е.

$$\{\beta_i\} - \sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. \le 0 . \qquad (5.11)$$

Отметим, что вершина с $\{\beta_i\} > 0$ не удовлетворяет последнему неравенству, т.е. она отсекается. Следовательно, выражение (5.11) представляет собой уравнение отсечения. Приведем его к каноническому виду

$$\sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. - x_{n+1} = \{\beta_i\} . \qquad (5.12)$$

Рассмотрим алгоритм решения задач целочисленного программирования (5.6)-(5.9), который сходится за конечное число шагов.

1). Решается задача линейного программирования (5.6) - (5.8) без ограничений на целочисленность. Если полученное решение целочисленно, то процесс заканчивается, если нет, то осуществляется переход на шаг 2.

2). Выделяется нецелочисленная компонента x_{si}, если таких компонент несколько, то выбирается та, у которой меньший номер i.

3). В систему ограничений вводится дополнительное (m+1)-е ограничение вида

$$\sum_{j \in j_q} \{\alpha_{ij}\}\, x_{qj}. - x_{n+1} = \{\beta_i\} ,$$

которое отсекает нецелочисленное решение и оставляет в допустимой области все целочисленные решения.

4). $m = m+1$; $n = n+1$. Переход на шаг 1.

Как это видно из алгоритма, размерность решаемой задачи растет на каждом шаге. Для устранения этого недостатка довольно часто используются алгоритмы, построенные на основе двойственной задачи линейного программирования (методы последовательного уточнения оценок).

5.4. Методы типа "ветвей и границ" в задачах дискретного программирования

Наряду с методами отсечения в дискретном программировании широко используются методы, основанные на последовательном анализе некоторых подмножеств допустимых решений, рассмотрении лишь тех подмножеств решений, которые оказываются перспективными по определенным признакам, и отбрасывании подмножеств решений, являющихся неперспективными. Такие переборные методы принято называть к о м б и н а т о р н ы м и методами [50, 83]. Одним из наиболее известных методов такого типа является метод ветвей и границ. Следует отметить, что под этим названием объединяется целая группа методов, построенных на идее анализа подмножеств решений, объединяемых в одном ветвлении ("ветви") на дереве поиска, и оценки перспективности данного ветвления путем вычисления некоторых оценок ("границ"). Такое построение алгоритма выбора наилучшего решения позволяет учесть специфические особенности конкретных моделей принятия решения и на этой основе повысить их эффективность.

Будем рассматривать применение метода ветвей и границ для задачи линейного целочисленного программирования, поставленной в стандартной форме (5.2)-(5.5).

$$L(x) = c^T x \to \max, \qquad (5.2)$$

$$A\,x \le b, \qquad (5.3)$$

$$x \ge 0, \qquad (5.4)$$

$$x \text{ - целое.} \qquad (5.5)$$

В результате решения данной задачи находится оптимальное решение x^* и $L^* = L(x^*)$. Несмотря на большое разнообразие применений метода ветвей и границ для различных моделей принятия решений, ряд операций является общими.

В е т в л е н и е. Так как значения переменных, описывающих решение в ограниченной области, принимают дискретные значения (пусть x_i i-я переменная принимает значения из $\{1,...,v_i\}$), то можно построить дерево поиска решений.

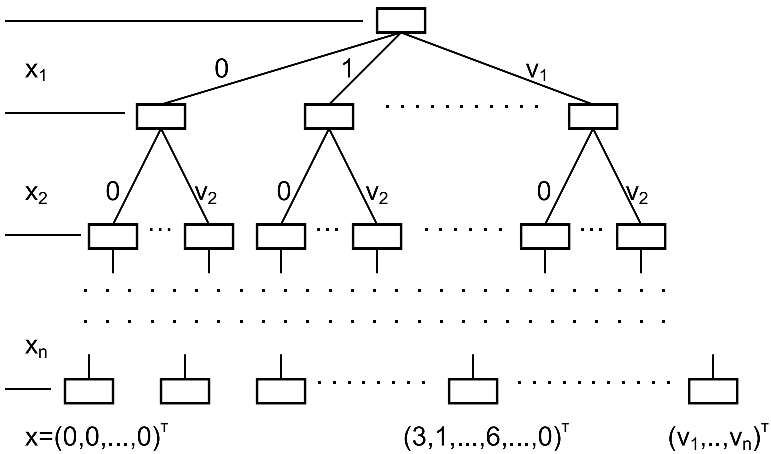


Рис. 5.2.

Висячие вершины последнего уровня такого дерева представляю собой различные решения. Часть этих решений может быть не допустима, т.к. для них не выполняются ограничения задачи. Все множество допустимых решений задачи (5.2)-(5.5) обозначим через Δ, тогда ветвления соответствуют разбиению Δ на подмножества можно представить в виде $\Delta = \Delta^1 \cup \Delta^2 \cup \ldots \cup \Delta^q$. Наибольшее распространение получили два вида ветвления.

Первый вид ветвления соответствует разбиению некоторого множества решений Δ^k на два непересекающихся подмножества. Например, $\Delta^k = \Delta^{k+1} \cup \Delta^{k+2}$, где $\Delta^{k+1} =\{ x \in \Delta \mid x_k \le 4\}$; $\Delta^{k+2} =\{ x \in \Delta \mid x_k > 4$.

Второй вид ветвления реализует покомпонентное ветвление, которое проводится путем фиксации значений переменных; например, $\Delta^k = \{x \in \Delta \mid x_k = 4\}$. В частности, на рис. 5.2 представлен именно этот вид ветвления.

В ы ч и с л е н и е г р а н и ц. Используется два вида границ.

Р е к о р д L^*_s - значение целевой функции оптимальное для всех допустимых решений, соответствующих ветвлению Δ^s. Рекорд L^*_s достигается на решении x^*_s, т.е. $L^*_s = L(x^*_s)$.

$$x^*_s = \arg \underset{x \in \Delta^s}{\mathrm{Max}}\ L(x)$$

О ц е н к а $g(\Delta^r)$. Для вычисления оценки $g_r = g(\Delta^r)$ решается оценочная задача. Как правило, это оптимизационная задача более простой структуры, чем исходная задача. К оценочной задаче обычно предъявляется два требования:

- если оценочная задача не имеет допустимых решений, то и исходная задача не имеет допустимых решений;
- оценка $g(\Delta^r)$ должна превышать значение рекорда для Δ^r, т.е. $g(\Delta^r) \geq L^*_r$.

Вычисление границ является основой для исключения из рассмотрения неперспективных подмножеств решений (ветвлений). Так, если для некоторого ветвления Δ^r оценка $g(\Delta^r)$ будет меньше, чем значение рекорда $L^*_s = L(x^*_s)$ для ветвления Δ^s, то это означает, что в ветвлении Δ^r не содержится целочисленных решений, лучших, чем решение $x^*_s \in \Delta^s$, и, следовательно, ветвление Δ^r можно исключить из дальнейшего рассмотрения.

Приведем один из возможных алгоритмов решения общей задачи целочисленного линейного программирования (5.2)-(5.5), основанных на методе ветвей и границ. В качестве оценочной задачи может выступать задача линейного программирования вида (5.2)-(5.4). В процессе решения задачи формируется список кандидатов Q, т.е. ветвлений, которые необходимо проанализировать; $Q = \{\Delta^1, \Delta^2, \ldots, \Delta^q\}$. В начале решения задачи $Q = \{\Delta\}$, $q = 1$; Δ - множество допустимых решений исходной задачи; значение рекорда $L^* = -\infty$. Итак, алгоритм заключается в выполнении последовательности итераций, каждая итерация состоит из четырех шагов.

Шаг 0.- Исходное состояние. Список кандидатов $Q = \{\Delta\}$; рекорд $L^* = -\infty$; $x^* = (-1,\ldots,-1)^T$.

Шаг 1.- Выбор кандидата. Из списка Q выбирается один из элементов, пусть это Δ^r - кандидат (ветвление).

Шаг 2.- Анализ кандидата. Если оценка g_r не известна, то решается оценочная задача и вычисляется оценка $g_r = g(\Delta^r)$.

Если $g_r \le L^*$ (L^* - текущий рекорд), то Δ^r исключается из списка кандидатов Q и производится переход на шаг 4.

Если $g_r > L^*$, тогда,

а) если при расчете оценки получено целочисленное решение x^*_r, то рекордом становится новое значение $L^*=g_r$, $x^*=x^*_r$. При этом ветвление Δ^r исключается из списка кандидатов Q и производится переход на шаг 4;

б) если x^*_r нецелочисленно, то выполняется шаг 3.

Шаг 3.- Ветвление. Согласно правилу ветвления производится разбиение Δ^r на два подмножества $\Delta^r = \Delta^{q+1} \cup \Delta^{q+2}$; при этом $\Delta^{q+1} \cap \Delta^{q+2} = \varnothing$. Далее Δ^r исключается из списка кандидатов Q, а Δ^{q+1} и Δ^{q+2} в этот список добавляются; q = q+2. Происходит переход на шаг 4.

Шаг 4.- Анализ списка кандидатов.

Если $Q = \varnothing$, тогда, процесс заканчивается. При этом,

если $L^* = -\infty$, то допустимых решений нет;

если $L^* > -\infty$, то решение x^* - оптимально.

Если $Q \neq \varnothing$, производится переход на шаг 1.

Конечность алгоритма, построенного по данной схеме, следует из конечности множества допустимых решений, содержащихся в конечном дереве поиска, и в последовательном вычеркивании кандидатов из списка.

Контрольные вопросы

1. В чем состоят особенности задач дискретного программирования по сравнению с задачами линейного и нелинейного программирования ?
2. Какие различают группы методов решения задач целочисленного программирования. В чем их особенность ?
3. Каким образом получить решение методом округления. Сколько вариантов округления может быть, как нужно выбирать один из вариантов ?
4. В каких ситуациях решение задач целочисленного программирования может быть получено непосредственно на основе использования непрерывных методов ?
5. В чем суть метода отсечений. Какие требования предъявляются к отсечению. Как его построить ?
6. В чем суть метода ветвей и границ; чем определяется его эффективность; какие задачи можно решать с его помощью ?

6. СПЕЦИАЛЬНЫЕ ЗАДАЧИ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Ряд задач математического программирования, имеющих важные практические приложения, обладают некоторыми специфическими особенностями, позволяющими строить вычислительные алгоритмы, более эффективные, чем те, которые используются в общих случаях. Это касается, прежде всего, задач линейного и дискретного программирования. Рассмотрим здесь некоторые наиболее известные и разработанные в алгоритмическом смысле задачи.

6.1. Транспортная задача

В 2.2 в качестве примера модели математического программирования рассматривалась классическая транспортная задача. Транспортная модель является частным случаем общей сетевой модели оптимизации, интерес к которым связан с тем, что в рамках терминологии сетевых (графовых) моделей описываются многие задачи принятия решений, связанные с управлением системами различной природы. Особенности математической структуры сетевых моделей позволяют строить эффективные вычислительные алгоритмы решения таких задач, и именно этот фактор позволяет выделить транспортные задачи в самостоятельный класс специальных задач, изучение которых имеет важные практические приложения в различных областях деятельности.

Представим транспортную модель в следующем виде [23, 30, 34. Имеется:

- множество поставщиков $A = \{A_i, i = 1,\dots, m\}$;
- множество потребителей $B = \{B_j, j = 1,\dots, n\}$.

Каждому поставщику соответствует объем имеющегося у него ресурса a_i, а каждому потребителю сопоставляется объем необходимого для него ресурса b_j. Предполагается, что каждый поставщик соединен с каждым потребителем, т.е. множество дуг такой сети $E = A \times B$. Задана функция $c: E \to R^1$, характеризующая некоторое качество (время, стоимость, и т.п.) перевозки по дуге $e_{ij} = (A_i, B_j) \in E$

В этом случае математической конструкцией, описывающей решение, может являться вектор

$$x = (x_{11}, x_{12}, \ldots, x_{1n}, x_{21}, \ldots, x_{2n}, \ldots, x_{mn})^{т} = \| x_{ij} \|,$$

компоненты которого x_{ij} характеризуют объем перевозок от поставщика A_i к потребителю B_j. Ограничения накладываются на объем продукции, вывозимой от i-го поставщика, и количество продукции, которую необходимо завести j-му потребителю. Целевая функция характеризует суммарные расходы на все такие перевозки.

Тогда математическая модель классической транспортной задачи имеет вид:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min \tag{6.1}$$

при ограничениях

$$\sum_{j=1}^{n} x_{ij} \le a_i, \ i = 1, \ldots, m, \tag{6.2}$$

$$\sum_{i=1}^{m} x_{ij} \ge b_j, \ j = 1, \ldots, n, \tag{6.3}$$

$$x_{ij} \ge 0, \quad i = 1, \ldots, m, \ j = 1, \ldots, n. \tag{6.4}$$

В ситуациях, когда имеется равенство суммарного ресурса у всех поставщиков и суммарных потребностей всех потребителей,

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \ , \tag{6.5}$$

неравенства в ограничениях транспортной модели приобретают вид равенств.

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min \tag{6.6}$$

$$\sum_{j=1}^{n} x_{ij} = a_i,\ i = 1,\ldots, m, \qquad (6.7)$$

$$\sum_{i=1}^{m} x_{ij} = b_j,\ j = 1,\ldots, n, \qquad (6.8)$$

$$x_{ij} \geq 0,\quad i = 1,\ldots, m,\ j = 1,\ldots,n. \qquad (6.9)$$

Такая задача называется стандартной (закрытой) транспортной задачей. Следует отметить, что произвольная транспортная задача может быть сведена к стандартной путем введения фиктивных поставщиков (потребителей), объем ресурсов (потребностей) которых равен разности левой и правой части в ограничениях (6.2) (или (6.3)).

Транспортная задача (6.6)-(6.9) описывается линейными ограничениями и линейной целевой функцией, следовательно, для ее решения могут использоваться стандартные методы линейного программирования (симплекс-метод, например). Размерность такой задачи равна $(m+n)\times(mn)$. Однако для решения транспортных задач был разработан специализированный алгоритм (метод потенциалов), учитывающий особенности данной задачи. Рассмотрим, в чем состоят эти особенности.

Каждой переменной x_{ij} соответствует перевозка от i-го поставщика к j-му потребителю. Соответственно тому, как все переменные x_{ij} можно разделить на два подмножества переменных, значения которых большие 0 и равные 0, можно разделить и все перевозки на запланированные и незапланированные.

Обозначим

- множество номеров пунктов производства; $M = \{1,...,m\}$;
- множество номеров пунктов потребления; $N = \{1,...,n\}$.

Тогда все возможные перевозки задаются множеством

$$J = M \times N.$$

В этом случае обозначим через

$S = \{(i,j)\in J \mid x_{ij}>0\}$ - множество запланированных (базисных) перевозок;

$Q = \{(i,j)\in J \mid x_{ij}=0\}$ - множество незапланированных (небазисных) перевозок.

Как в любой задаче линейного программирования число базисных переменных опорного решения не превышает числа ограничений, так и здесь число базисных перевозок не превышает $|S| \leq (m + n)$. Вместе с тем специфика транспортной задачи такова, что любую строку матрицы условий можно выразить как линейную комбинацию остальных, т.е. ранг системы ограничений равен (m+n-1). Действительно, матрица условий имеет вид

$$A = \begin{array}{ll} \left.\begin{array}{l} 1\;1\ldots1\;0\;0\ldots0\;0\;.\;.\;.\;.\;.\;0\;0\ldots0 \\ 0\;0\ldots0\;1\;1\ldots1\;0\;.\;.\;.\;.\;.\;0\;0\ldots0 \\ .\;. \\ .\;. \\ 0\;0\ldots0\;0\;0\ldots0\;0\;.\;.\;.\;.\;.\;1\;1\ldots1 \end{array}\right\} & m\text{-строк} \\ \left.\begin{array}{l} 1\;0\ldots0\;1\;0\ldots0\;1\;.\;.\;.\;.\;.\;1\;0\ldots0 \\ 0\;1\ldots0\;0\;1\ldots0\;0\;.\;.\;.\;.\;.\;0\;1\ldots0 \\ .\;. \\ 0\;0\ldots1\;0\;0\ldots1\;0\;.\;.\;.\;.\;.\;0\;0\ldots1 \end{array}\right\} & n\text{-строк} \end{array}$$

Тогда последняя строка A является линейной комбинацией первых (m+n-1) строк:

$$a^{т}_{m+n} = \sum_{i=1}^{m} a^{т}_{i} - \sum_{j=1}^{n-1} a^{т}_{j}$$

Следовательно, det A = 0 и в базисе S содержится (m+n-1) переменная, т.е.rang A = m+n-1. Тогда в любой транспортной задаче может быть запланировано только (m+n-1) перевозка.

Следующая особенность транспортной задачи заключается в том, что, как отмечалось в 5.2, матрица условий удовлетворяет достаточным условиям абсолютной унимодулярности. Тогда многогранник решений транспортной задачи является целочисленным при любом целочисленном векторе ограничений, а следовательно, все опорные решения и оптимальное решение будут целочисленны.

Для решения транспортной задачи был разработан специализированный метод - метод потенциалов [45], суть которого состоит в том, что, начиная с некоторого опорного решения (множество запланированных перевозок), последовательно анализируется его оптимальность. Если решение не оптимально, то находится перевозка, которую следует запланировать (ввести в число базисных), и находится перевозка, которую следует вывести из базиса. Для расчетов по методу потенциалов используется таблица, элементами которой являются стоимости перевозок c_{ij}, i = 1,…, m, j = 1,…,n. Таким образом, специализированный метод потенциалов более эффективен для решения транспортных задач, т.к. он оперирует с задачей размерности $m\times n$, в отличие от симплекс-метода, предназначенного для решения общей задачи линейного программирования, который оперирует с задачей размерности $(m+n)\times(mn)$.

6.1.1. Построение исходного опорного плана транспортной задачи

Построение исходного опорного плана (решения) транспортной задачи заключается в нахождении множества перевозок, позволяющих вывести все товары от поставщиков и обеспечивающих всех потребителей. Такие перевозки характеризуются переменными

$$\{ x_{ij} > 0 \mid (i,j)\in S \subseteq J = M\times N \} .$$

Рассмотрим два способа построения исходного опорного плана перевозок. При расчетах опорного решения используется два вектора

$A^k = (A^k{}_1,\ldots,A^k{}_m)$ – вектор, характеризующий использование ресурсов, имеющихся у поставщиков;

$B^k = (B^k{}_1,\ldots,B^k{}_n)$ – вектор, характеризующий текущее удовлетворение запросов потребителей.

Значения компонент данных векторов изменяются на каждой на k-ой итерации метода.

Результаты расчетов при построении исходного опорного плана заносятся в таблицу вида

c_{11} x_{11}	………	c_{1n} x_{1n}	a_1
……	c_{ij} x_{ij}	……	……
c_{m1} x_{m1}	………	c_{mn} x_{mn}	a_m
b_1	………	b_n	a_i b_j

Здесь a_i, b_j, c_{ij} - исходные данные транспортной задачи: a_i - количество ресурса i-го поставщика; b_j - запросы j-го потребителя; c_{ij} и x_{ij} – соответственно стоимость перевозки единицы груза и объем перевозки от i–го поставщика j–му потребителю.

<u>Метод северо-западного угла.</u>

Поиск опорного плана начинается с северо-западного (правого верхнего) угла матрицы перевозок.

0). Исходное состояние: номер итерации $k = 1$;

$A^1 = (A^1{}_1,...,A^1{}_m)$, $A^1{}_i = a_i$, $i \in M$;

$B^1 = (B^1{}_1,...,B^1{}_n)$, $B^1{}_j = b_j$, $j \in N$;

текущая перевозка $(i,j) = (1,1)$.

Каждая k-я итерация состоит в выполнении 3-х шагов:

а). Вычисляется объем перевозки (i,j)

$x_{ij} = \min \{A^k{}_i, B^k{}_j\}$.

Пересчитываются значения компонент векторов A и B:

$A_i^{k+1} = A^k{}_i - x_{ij}$;

$B_j^{k+1} = B^k{}_j - x_{ij}$.

б). Если $A_i^{k+1} = 0$, $i \in M$ и $B_j^{k+1} = 0$, $j \in N$, то процесс окончен, т.е. все ресурсы поставщиков исчерпаны, и все запросы потребителей удовлетворены. Если это не так, то определяется очередная перевозка: если $A^k{}_i \geq B^k{}_j$, то $j = j + 1$;

если $A^k{}_i < B^k{}_j$, то $i = i + 1$.

в). $k = k+1$. Переход на шаг а.

<u>Рассмотрим пример.</u>

Пусть производится перевозка грузов от 3-х поставщиков ($a_1=11$; $a_2=11$; $a_3=8$) к 4-м потребителям ($b_1=5$; $b_2=9$; $b_3=9$; $b_4=7$). Необходимо построить исходный опорный план перевозок, если заданы стоимости перевозок единицы груза (c_{ij}).

0). Исходное состояние: $k = 1$;

$A^1 = (11, 11, 8)$;

$B^1 = (5, 9, 9, 7)$;

текущая перевозка $(i,j) = (1,1)$.

7 5	8 6	5	3	11
2	4 3	5 8	9	11
6	3	1 1	2 7	8
5	9	9	7	a_i b_j

Итерация 1. $k = 1$;

$x_{11} = 5$; $A^2 = (6, 11, 8)$;

$B^2 = (0, 9, 9, 7)$;

$A^1{}_1 \geq B^1{}_1$ - $j = j +1 = 2$.

Итерация 2. $k = 2$;

$x_{12} = 6$; $A^3 = (0, 11, 8)$;

$B^3 = (0, 3, 9, 7)$;

$A^2{}_1 < B^2{}_2$ - $i = i +1 = 2$.

Итерация 3. $k = 3$;

$x_{22} = 3$; $A^4 = (0, 8, 8)$;

$B^4 = (0, 0, 9, 7)$;

$A^3{}_2 \geq B^3{}_2$ - $j = j +1 = 3$.

Итерация 4. k = 4;

$x_{23} = 8$; $A^5 = (0, 0, 8)$;

$B^5 = (0, 0, 1, 7)$;

$A^4{}_2 < B^4{}_3$ - $i = i + 1 = 3$.

Итерация 5. k = 5;

$x_{33} = 1$; $A^6 = (0, 0, 7)$;

$B^6 = (0, 0, 0, 7)$;

$A^5{}_2 \geq B^5{}_3$ - $j = j + 1 = 4$.

Итерация 6. k = 6;

$x_{34} = 7$; $A^7 = (0, 0, 0)$;

$B^7 = (0, 0, 0, 0)$;

$\forall A^7{}_i = 0$; $\forall B^7{}_j = 0$; - конец.

Значение целевой функции (стоимость перевозок) $L = 7\times5 + 8\times6 + 4\times3 + 5\times8 + 1\times1 + 2\times7 = 150$. Исходный опорный план перевозок построен, однако в процессе поиска исходного опорного решения не учитывалась стоимость перевозок.

<u>Метод минимального элемента.</u>

Представляется целесообразным стоимость перевозок учитывать уже на этапе построения исходного плана. Для этого предназначен метод минимального элемента.

0). Исходное состояние: номер итерации k = 1;

$A^1 = (A^1{}_1,\dots,A^1{}_m)$, $A^1{}_i = a_i$, $i \in M$;

$B^1 = (B^1{}_1,\dots,B^1{}_n)$, $B^1{}_j = b_j$, $j \in N$;

в качестве текущей перевозки выбирается перевозка с минимальной стоимостью

$$(i, j) = \arg \min_{\substack{r\in M \\ s\in N}} c_{rs}$$

Каждая k-я итерация состоит в выполнении 3-х шагов:

а). Вычисляется объем перевозки (i,j)

$x_{ij} = \min \{A^k{}_i, B^k{}_j\}$.

Пересчитываются значения компонент векторов A и B:

$A_i^{k+1} = A^k{}_i - x_{ij}$;

$B_j^{k+1} = B^k{}_j - x_{ij}$.

б). Если $A_i^{k+1} = 0, i\in M$ и $B_j^{k+1} = 0, j\in N$, то процесс окончен, т.е. все ресурсы поставщиков исчерпаны, и все запросы потребителей удовлетворены. Если это не так, то определяется очередная перевозка:

$$(i, j) = \arg \min_{\substack{r \in M \,|\, A_r^{k+1} > 0 \\ s \in N \,|\, B_s^{k+1} > 0}} c_{rs}$$

в). $k = k+1$. Переход на шаг а.

Построим исходный опорный план перевозок предыдущего примера с использованием метода минимального элемента.

7	8 3	5 1	3 7	11
2 5	4 6	5	9	11
6	3	1 8	2	8
5	9	9	7	a_i b_j

0). Исходное состояние: $k = 1$;
$A^1 = (11, 11, 8)$;
$B^1 = (5, 9, 9, 7)$;
текущая перевозка $(i,j)=(3,3)$.

Итерация 1. $k = 1$;
$x_{33} = 8$; $A^2 = (11, 11, 0)$;
$B^2 = (5, 9, 1, 7)$;
$(i,j)=(2, 1)$.

Итерация 2. $k = 2$;
$x_{21} = 5$; $A^3 = (11, 6, 0)$;
$B^3 = (0, 9, 1, 7)$;
$(i,j)=(1, 4)$.

Итерация 3. $k = 3$;
$x_{14} = 7$; $A^4 = (4, 6, 0)$;
$B^4 = (0, 9, 1, 0)$;
$(i,j)=(2, 2)$.

Итерация 4. $k = 4$;
$x_{22} = 6$; $A^5 = (4, 0, 0)$;
$B^5 = (0, 3, 1, 0)$;
$(i,j)=(1, 3)$.

Итерация 5. $k = 5$;
$x_{13} = 1$; $A^6= (3, 0, 0)$;
$B^6 = (0, 3, 0, 0)$;
$(i,j)=(1, 2)$.

Итерация 6. $k = 6$;
$x_{12} = 3$; $A^7= (0, 0, 0)$;
$B^7 = (0, 0, 0, 0)$;
$\forall A^7{}_i = 0$; $\forall B^7{}_j = 0$; - конец.

Значение целевой функции (стоимость перевозок) $L = 8\times3 + 5\times1 + 3\times7 + 2\times5 + 4\times6 + 1\times8 = 92$. Исходный опорный план, найденный методом минимального элемента, значительно превышает по качеству план перевозок, найденный методом северо-западного угла.

6.1.2. Метод потенциалов

(Канторович Л.В., Гавурин М.К., 1940)

Транспортная задача заключается в необходимости планирования перевозок некоторого груза от поставщиков (производителей) к потребителям. Отдельную перевозку можно характеризовать

$$A_i \xrightarrow[x_{ij}]{c_{ij}} B_j, \quad i \in M = \{1,\dots,m\};\ j \in N = \{1,\dots,n\}.$$

Здесь c_{ij} - стоимость перевозки (задана в исходных данных); x_{ij} - объем перевозки (необходимо найти в результате планирования).

Математическая модель задачи описывается выражениями (6.6)-(6.9). Решение задачи, двойственной к задаче (6.6)-(6.9), будет иметь вид (см.3.1)

$$\mu = (U_1,\dots,U_i,\dots,U_m,V_1,\dots,V_j,\dots,V_n)^T,$$

где U_i характеризует эффективность повышения количества ресурсов a_i и называется потенциалом i-го пункта производства, а V_j характеризует эффективность повышения величины спроса b_j и называется потенциалом j-го пункта потребления. Тогда разность потенциалов $(V_j - U_i)$ характеризует эффективность перевозки (i,j). В этом случае, если $(V_j - U_i) > c_{ij}$, то перевозку (i,j) целесообразно запланировать, если $(V_j - U_i) < c_{ij}$, то перевозка (i,j) нецелесообразна.

Опорное решение $x = \|x_{ij}\| = \|x_s\ x_q\|$, где $x_s > 0$ - вектор базисных перевозок (запланированных); и $x_q = 0$ - вектор небазисных (незапланированных) перевозок. Все перевозки, определяемые опорным решением x, характеризуются множеством $J = M \times N$, причем $J = S \cup Q$,

где S - базисные перевозки $S = \{\, (i,j) \in J \mid x_{ij} \ge 0 \,\}$.

Q - небазисные перевозки $Q = \{\, (i,j) \in J \mid x_{ij} = 0 \,\}$.

Очевидно, что если $(i,j) \in S$, то $(V_j - U_i) = c_{ij}$, с другой стороны, если выполняется

$$\forall (i,j) \in Q,\ (V_j - U_i) \le c_{ij}\,, \tag{6.10}$$

то нет перевозок, которые целесообразно запланировать, следовательно, решение оптимально. В то же время, если

$$\exists(\alpha,\beta) \in Q, \ (V_\beta - U_\alpha) > c_{\alpha\beta}, \qquad (6.11)$$

то перевозку $K = (\alpha,\beta)$ целесообразно ввести в число базисных, тогда $x_{\alpha\beta} > 0$. Однако, в базисе S (соответственно, в опорном решении) не может содержаться более $(m+n-1)$ переменных, тогда некоторую перевозку $R = (\gamma,\delta)$ следует вывести из базиса S.

Алгоритм решения транспортной задачи, основанный на методе потенциалов, представляет собой итерационный алгоритм [45]. На каждой итерации выполняются три шага:

а). Проверяется условие (6.10) и, если оно выполняется, то текущее решение оптимально. Если (6.10) не выполняется, то выбирается перевозка $K=(\alpha,\beta)$, которая вводится в число базисных.

б). Выявляется перевозка $R=(\gamma,\delta)$, которую следует вывести из базиса.

в). Производится пересчет текущего базисного решения, и вычисляется значение целевой функции.

Рассмотрим содержание описанных шагов.

а). Анализ оптимальности текущего решения. Определение перевозки, вводимой в базис.

♦ Вычисляются потенциалы U_i, V_j из системы уравнений

$$V_j - U_i - c_{ij} = 0, \ \forall(i,j) \in S,$$

причем $U_1 = \text{const}$, например, $U_1 = 0$.

♦ Рассчитывается невязка

$$\eta(\alpha,\beta) = \max_{(i,j)\in Q} (V_j - U_i - c_{ij}).$$

Если $\eta(\alpha,\beta) \le 0$, то решение оптимально.

Если $\eta(\alpha,\beta) > 0$, то перевозку $K = (\alpha,\beta)$ целесообразно ввести в базис S.

б). Нахождение перевозки, выводимой из базиса.

Строится ε-маршрут. Для некоторого базиса S запланированные перевозки характеризуются графом.

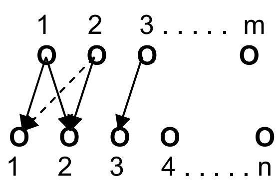
Например,

$S = \{(1,1),(1,2),(2,2),(3,3),... \}$

Пусть $K = (2,1)$, тогда

ε-маршрут имеет вид

ε: (2,1)-(1,1)-(1,2)-(2,2).



Т.е. для того, чтобы произвести перевозку (2,1), необходимо изменить перевозки (1,1), (1,2), (2,2). Причем те перевозки, которые в ε-маршруте проходятся в прямом направлении, должны быть увеличены на величину ε, те же перевозки, которые проходятся в обратном направлении, должны быть уменьшены на ε. Тогда

$$\varepsilon:\ \underset{+}{(2,1)}-\underset{-}{(1,1)}-\underset{+}{(1,2)}-\underset{-}{(2,2)}.$$

В качестве ε берется минимальное значение из тех перевозок, которые должны быть сокращены. Т.е.

$$\varepsilon = \min\ x^{-}_{ij},\ (i,j) \in S$$

Перевозка $R = (\gamma,\delta)$, на которой достигается ε, будет равна 0, т.е. фактически она выводится из базиса S.

в). Пересчет опорного решения.

♦ Новый базис S' вычисляется как

$$S' = S \cup K \setminus R.$$

♦ В новом опорном решении x' изменяется значение перевозок на ε-маршруте по правилу

$$x_{ij}'^{-} = x_{ij}^{-} - \varepsilon$$

$$x_{ij}'^{+} = x_{ij}^{+} + \varepsilon$$

♦ Вычисляется новое значение целевой функции

$$L(x') = L(x) - \eta\varepsilon.$$

Все данные результатов счета на каждой итерации представляются в таблице вида

U_i \ V_j	V_1	....	V_n	
U_1	c_{11} x_{11}	....	c_{1n} x_{1n}	a_1
......	......	c_{ij} x_{ij}	......	......
U_m	c_{m1} x_{m1}	....	c_{mn} x_{mn}	a_m
	b_1	....	b_n	a_i b_j

Рассмотрим пример из п.6.1.1. Возьмем в качестве исходного опорного решения решение, полученное методом наименьшего элемента. $\eta(i,j) = V_j - U_i - c_{ij}$

Итерация 1.

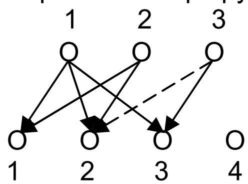
а).

♦ Вычисляются потенциалы
$V_j - U_i - c_{ij} = 0, \ \forall(i,j) \in S$,

♦ Рассчитываются невязки
$\eta(i,j) = V_j - U_i - c_{ij}$ для $(i,j) \in Q$
$\eta(1,1) = -1$; $\eta(3,1) = -4$;
$\eta(2,3) = -4$; $\eta(3,2) = 1$;
$\eta(2,4) = -10$; $\eta(3,4) = -3$;
$\eta = 1$; $K = (3,2)$;

V_j / U_i	6	8	5	3	
0	7	8 3-	5 1+	3 7	11
4	2 5	4 6	5	9	11
4	6	3 ε	1 8-	2	8
	5	9	9	7	a_i / b_j

б). Строится ε-маршрут



ε: (3,2)-(1,2)-(1,3)-(3,3)
+ - + -
$\varepsilon = \min\{3, 8\} = 3$; $R = (1,2)$.

в). Производится пересчет таблицы.
$x_{32}^{-} = 3$; $x_{12}^{-} = 0$; $x_{13}^{-} = 4$; $x_{33}^{-} = 5$;
$L = 92 - 1 \times 3 = 89$.

Итерация 2.

а).

♦ Вычисляются потенциалы
$V_j - U_i - c_{ij} = 0, \ \forall(i,j) \in S$,

♦ Рассчитываются невязки
$\eta(i,j) = V_j - U_i - c_{ij}$ для $(i,j) \in Q$
$\eta(1,1) = -2$; $\eta(2,4) = -9$;
$\eta(1,2) = -1$; $\eta(3,1) = -5$;
$\eta(2,3) = -3$; $\eta(3,4) = -3$;
$\eta = -1$;

V_j / U_i	5	7	5	3	
0	7	8	5 **4**	3 **7**	11
3	2 **5**	4 **6**	5	9	11
4	6	3 **3**	1 **5**	2	8
	5	9	9	7	a_i / b_j

- решение оптимально.
$x_{13}^{-} = 4$; $x_{14}^{-} = 7$; $x_{21}^{-} = 5$; $x_{22}^{-} = 6$; $x_{32}^{-} = 3$; $x_{33}^{-} = 5$; $L = 89$.

Отметим, что если в качестве исходного опорного решения использовать план, полученный методом северо-западного угла (см.6.1.1), то в этом случае, то же самое оптимальное решение будет получено на пятой итерации метода потенциалов.

6.2. Транспортная задача в сетевой постановке

Пусть имеется множество A пунктов производства, транспортировки и потребления, соединенных транспортной сетью дорог (участками пути)

$A = A^p \cup A^t \cup A^k$,

здесь

A^p - множество пунктов производства продукции;

A^t - множество пунктов транспортировки продукции;

A^k - множество пунктов потребления продукции;

Если имеется всего n пунктов (узлов) транспортной сети, то

$A = \{A_1,..., A_n\} = \{ A_i, i \in I\}$, здесь $I = \{1,2,..., n\}$.

Тогда и множество номеров пунктов I можно представить в виде

$I = I^p \cup I^t \cup I^k$,

где

$I^p = \{P^1,...,P^p\}$; $I^t = \{T^1,...,T^t\}$; $I^k = \{K^1,...,K^k\}$, $n = p + t + k$.

На множестве пунктов (узлов) A строится транспортная сеть Г, которая задается

$\Gamma = (A, E, C)$,

где $E \subseteq A \times A$ - множество дуг графа транспортной сети (задает возможные перевозки);

$E = \{(A_i, A_j),\ i,j \in I\} = \{e_q, q=1,...,z \}$,

то есть $e_q = (A_i, A_j)$ и $|E| = z$;

$C: E \to R^1$ – функция, описывающая качество перевозок.

Обозначим множество возможных перевозок через J, тогда

$J = \{ (i,j) \mid (A_i, A_j) \in E\}$

Каждому пункту A_i сопоставляется величина потребления продукции a_i, причем, если $A_i \in A^p$, то $a_i > 0$

если $A_i \in A^k$, то $a_i < 0$

если $A_i \in A^t$, то $a_i = 0$.

Далее будем полагать, что $\sum_{i=1}^{n} a_i = 0$. Если это условие не выполняется, то его можно обеспечить путем введения фиктивного пункта производства (или потребления).

В этих условиях необходимо найти план, минимизирующий суммарную стоимость перевозок.

Для математической постановки задачи необходимо, прежде всего, определить конструкцию, которой будет описываться решение (план перевозок). По аналогии, как и в ситуации классической транспортной задачи (см. 6.1), план перевозок будем описывать вектором x

$$x = \left\| x_{ij}.\ (i,j) \in J \right\|. \qquad (6.12)$$

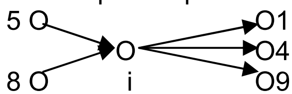
Компоненты x_{ij} характеризуют объем перевозок из узла A_i в узел A_j. Ограничения на возможные перевозки накладываются структурой транспортной сети, объемом продукции, имеющейся на пунктах производства, и объемом продукции необходимой пунктам потребления. Целевая функция характеризует суммарные расходы на все такие перевозки.

Обозначим через I_i^- - множество узлов сети (пунктов), в которые могут осуществляться перевозки из i-го узла; соответственно, I_i^+- множество узлов сети, из которых могут осуществляться перевозки в i-ый узел;

$$I_i^- = \{ j \in I \mid (i,j) \in J\} = \{ j \in I \mid (A_i, A_j) \in E\}, \qquad (6.13)$$

$$I_i^+ = \{ j \in I \mid (j,i) \in J\} = \{ j \in I \mid (A_j, A_i) \in E\}. \qquad (6.14)$$

Так например, если фрагмент транспортной сети имеет вид



то $I_i^- = \{1, 4, 9\}$; $I_i^+ = \{5, 8\}$.

Тогда математическая модель транспортной задачи в сетевой постановке имеет вид:

$$\sum_{(i,j)\in J} c_{ij}\, x_{ij} \to \min \qquad (6.15)$$

при ограничениях

$$\sum_{j\in I_i^-} x_{ij} - \sum_{j\in I_i^+} x_{ji} = a_i,\ \ i = 1,\ldots, n, \qquad (6.16)$$

$$x_{ij} \geq 0,\ \ (i,j) \in J. \qquad (6.17)$$

Число ограничений такой задачи равно числу узлов транспортной сети (n), а число переменных равно числу дуг графа сети (z). Задача (6.15)-(6.17) относится к классу задач линейного программирования и может решаться с использованием стандартных методов. Однако для решения такой задачи был разработан модифицированный метод потенциалов, учитывающий ее специфику.

Введем некоторые понятия и дадим определения, которые будут использоваться в дальнейшем при изложении алгоритмов.

♦ П у т е м W_{ij} из узла i в узел j транспортной сети будем называть упорядоченную последовательность перевозок таких, что конец одной из них является началом другой

$$W_{ij} = \{ (i,j_1), (j_1,j_2), (j_2,j_3), \dots, (j_{k-1},j)\},$$

количество элементов (k) в этой последовательности называется д л и н о й п у т и; $|W_{ij}| = k$.

Каждому пути W_{ij} может быть поставлена в соответствие стоимость перевозок $C(W_{ij})$ по этому пути

$$C(W_{ij}) = \sum_{(r,s)\in W_{ij}} c_{rs}\, x_{rs}$$

♦ Путем м и н и м а л ь н о й стоимости из узла i в узел j будем называть путь, W^*_{ij} такой, что

$$W^*_{ij} = \arg \min_{\{W_{ij}\}} C(W_{ij})$$

$C(W^*_{ij})$ – минимальная стоимость перевозок из узла i в узел j.

Метод потенциалов позволяет оценить оптимальность текущего плана перевозок и, если план не является оптимальным, то позволяет определить пути улучшения такого плана. Таким образом, для начала работы алгоритма необходимо найти исходный опорный план.

<u>Построение опорного плана.</u>

Рассмотрим модификацию метода минимального элемента применительно к транспортной задаче в сетевой постановке.

0). Исходное состояние: номер итерации $k = 1$;

$$A^1 = (A^1_1,\dots,A^1_n),\ A^1_i = |a_i|,\ i \in I;$$

в качестве текущего пути (i,j) выбирается путь минимальной стоимости

$$W_{ij} = \arg \min_{\substack{r\in I^p \\ s\in I^k}} C(W_{rs}),$$

здесь I^p - множество номеров пунктов производства продукции;

I^k - множество номеров пунктов потребления продукции.

Каждая k-я итерация состоит в выполнении 3-х шагов:

а). Пересчитывается объем перевозок по пути W_{ij}

$$g_{ij} = \min \{A^k_i, A^k_j\},$$

$$x'_{rs} = x_{rs} + g_{ij},\ (r,s) \in W_{ij}$$

Пересчитываются значения компонент вектора A :

$$A_i^{k+1} = A^k_i - g_{ij};$$

$$A_j^{k+1} = A^k_j - g_{ij}.$$

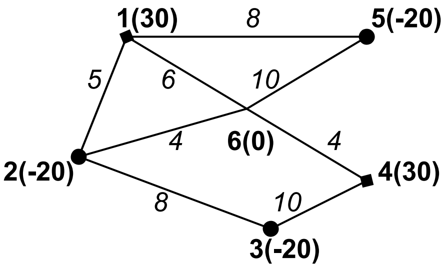
б). Если $A_i^{k+1} = 0$, $i \in I$, то процесс окончен, т.е. все ресурсы поставщиков исчерпаны, и все запросы потребителей удовлетворены. Если это не так, то определяется очередной путь:

$$W_{ij} = \arg \min_{\substack{r \in I^p \,|\, A_r^{k+1} > 0 \\ s \in I^k \,|\, A_s^{k+1} > 0}} C(W_{rs})$$

в). $k = k+1$. Переход на шаг а.

Пример.

Пусть задана транспортная сеть на множестве из шести узлов. При этом $I^p = \{1, 4\}$; I^t $\{6\}$; $I^k = \{2, 3, 5\}$; a_1=30; a_2=-20; a_3=-20; a_4=30; a_5=-20; a_6=0. Граф транспортной сети и стоимости перевозок задаются в следующем виде:



0). Исходное состояние: $k = 1$;

$A^1 = (30, 20, 20, 30, 20, 0)$;

текущий путь $(i,j)=(1,2)$.

Итерация 1. $k = 1$;

$x_{12} = 20$; $A^2 = (10, 0, 20, 30, 20, 0)$;

$(i,j)=(1, 5)$.

Итерация 2. $k = 2$;

$x_{15} = 10$; $A^3 = (0, 0, 20, 30, 10, 0)$;

$(i,j)=(4, 3)$.

Итерация 3. $k = 3$;

$x_{43} = 20$; $A^4 = (0, 0, 0, 10, 10, 0)$;

$(i,j)=((4,6), (6,5))$.

Итерация 4. $k = 4$;

$x_{46} = x_{65} = 10$; $A^5 = (0, 0, 0, 0, 0, 0)$;

$\forall A^5_i = 0$; - конец.

Исходный опорный план построен:

$x_{12} = 20$; $x_{15} = 10$; $x_{43} = 20$; $x_{46} = x_{65} = 10$;
значение целевой функции (стоимость перевозок)

$$L = 5\times20 + 8\times10 + 10\times20 + 4\times10 + 10\times10 = 520.$$

<u>Метод потенциалов.</u>

В (6.1) был рассмотрен метод потенциалов, предназначенный для решения транспортных задач. Рассмотрим особенности применения этого метода для решения транспортной задачи в сетевой постановке [78]. Математическая модель задачи описывается выражениями (6.15)-(6.17). Решение двойственной задачи будет иметь вид

$$\mu = (V_1,\ldots,V_j,\ldots,V_n)^т,$$

где V_j называют потенциалом j -го узла сети, и он характеризует эффективность изменения a_j. Тогда разность потенциалов $|V_j-V_i|$ характеризует эффективность перевозки.

Все перевозки J, определяемые опорным решением $x= \| x_s \; x_q \|$, можно разбить на два подмножества $J = S \cup Q$,

где S - базисные перевозки $S = \{ (i,j)\in J \mid x_{ij} > 0 \}$.

Q - небазисные перевозки $Q = \{ (i,j)\in J \mid x_{ij} = 0 \}$.

Если $(i,j) \in S$, то $(V_j - V_i) = c_{ij}$.

Если $\forall(i,j) \in Q$, $|V_j - V_i| < c_{ij}$, то нет перевозок, которые целесообразно запланировать, следовательно, решение оптимально.

Если $\exists(\alpha,\beta) \in Q$, $|V_\beta - U_\alpha| > c_{\alpha\beta}$, тогда, если $(V_\beta - U_\alpha)>0$, то перевозку $K = (\alpha,\beta)$ целесообразно ввести в число базисных, если $(V_\beta - U_\alpha)<0$, то в базис следует ввести перевозку $K = (\beta,\alpha)$.

Рассмотрим алгоритм решения транспортной задачи в сетевой постановке.

а). Анализ оптимальности текущего решения. Определение перевозки, вводимой в базис.

♦ Вычисляются потенциалы V_j из системы уравнений

$$V_j - V_i - c_{ij} = 0, \quad \forall(i,j) \in S,$$

причем V_1 = const, например, $V_1 = 0$.

♦ Рассчитывается невязка

$$\eta(\alpha,\beta) = \max_{(i,j)\in Q} (|V_j - V_i| - c_{ij}).$$

Если $\eta(\alpha,\beta)\le0$, то решение оптимально. Процесс заканчивается.

Если $\eta(\alpha,\beta)>0$, то определяется перевозка, вводимая в базис:

если $(V_\beta - V_\alpha) > 0$, то перевозка $K = (\alpha,\beta)$ вводится в базис S;

если $(V_\beta - V_\alpha) < 0$, то перевозка $K = (\beta,\alpha)$ вводится в базис S.

б). Нахождение перевозки, выводимой из базиса.

- Строится ε-маршрут.
- Рассчитывается ε :

$$\varepsilon = \min\ x^-_{ij},\ (i,j) \in S$$

Перевозка $R = (\gamma,\delta)$, на которой достигается ε, выводится из базиса S.

в). Строится новый граф перевозок

- Новый базис S' вычисляется как

$$S' = S \cup K \setminus R.$$

- В новом опорном решении x' изменяется значение перевозок на ε-маршруте по правилу

$$x_{ij}'^{-} = x_{ij}^{-} - \varepsilon$$

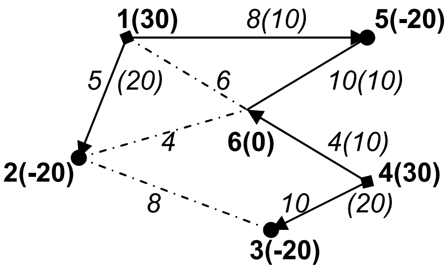
$$x_{ij}'^{+} = x_{ij}^{+} + \varepsilon$$

- Вычисляется новое значение целевой функции

$$L(x') = L(x) - \eta\varepsilon.$$

Пример.

Применим рассмотренный алгоритм для улучшения опорного решения, полученного методом минимального элемента в предыдущем примере. Опорное решение имеет вид:



Значение целевой функции $L = 520$.

Итерация 1.

а).

- Вычисляются потенциалы

$V_j - U_i - c_{ij} = 0,\ \forall(i,j) \in S,\ V_1 = 0.$

Тогда $V_1 = 0$; $V_2 = 5$; $V_3 = 4$; $V_4 = -6$; $V_5 = 8$; $V_6 = -2$.

- Рассчитываются невязки

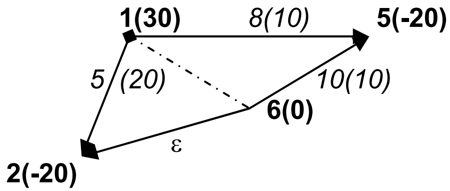
$\eta(i,j) = (|V_j - V_i| - c_{ij})$ для $(i,j) \in Q$

$\eta(1,6) = 2 - 6 = -4;$

$\eta(2,6) = 7 - 4 = +3$;
$\eta(2,3) = 1 - 8 = -7$;
$\eta(2,6) = \max \eta(i,j) = 3$, т.к. $(V_6 - V_2) = -7 < 0$, то $K = (6,2)$.

б).

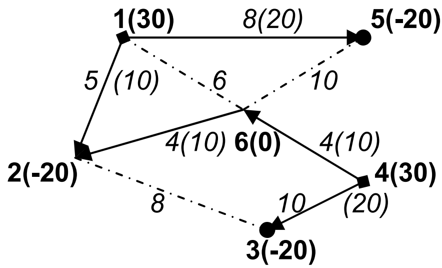
♦ Строится ε-маршрут



(6,2) - (1,2) - (1,5) - (6,5)
  +   -   +   -

♦ Рассчитывается $\varepsilon = \min \{20, 10\} = 10$.

в). Строится новый граф перевозок



Значение целевой функции $L = 520 - 3 \times 10 = 490$.

Итерация 2.

а).

♦ Вычисляются потенциалы
$V_j - U_i - c_{ij} = 0, \ \forall(i,j) \in S, \ V_1 = 0$.
Тогда $V_1 = 0$; $V_2 = 5$; $V_3 = 7$; $V_4 = -3$; $V_5 = 8$; $V_6 = 1$.

♦ Рассчитываются невязки
$\eta(i,j) = (|V_j - V_i| - c_{ij})$ для $(i,j) \in Q$
$\eta(1,6) = 1 - 6 = -5$;
$\eta(2,3) = 2 - 8 = -6$;
$\eta(5,6) = 7 - 10 = -3$;
$\eta(5,6) = \max \eta(i,j) = -3 < 0$, следовательно, решение, полученное на предыдущей итерации оптимально.

Таким образом, оптимальный план перевозок имеет вид: $x_{12} = 10$; $x_{15} = 20$; $x_{43} = 20$; $x_{46} = 10$; $x_{62} = 10$.

Суммарная стоимость перевозок составляет 490.

6.3. Задача о назначениях

Частным случаем классической транспортной задачи (6.1)-(6.4) является задача о назначениях, которая помимо того, что сама имеет важное прикладное значение, часто используется, как вспомогательная при решении дискретных оптимизационных задач различного рода. Практическая суть задачи, из которой и возникло ее название, заключается в том, что имеется (n) должностей, на которые могут быть назначены (n) кандидатов, причем назначение некоторого i–го кандидата на j-ю должность связано с расходами равными c_{ij}. Необходимо так распределить кандидатов на должности, чтобы суммарные расходы были минимальны.

В этих условиях решение (распределение кандидатов) можно описать вектором x с компонентами x_{ij}, которые принимают значение 1 или 0 в зависимости от того, назначается i–ый кандидат на j–ю должность, или нет. Математическая модель такой задачи имеет вид

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min \qquad (6.18)$$

при ограничениях

$$\sum_{j=1}^{n} x_{ij} = 1, \ i = 1,\ldots, n, \qquad (6.19)$$

$$\sum_{i=1}^{n} x_{ij} = 1, \ j = 1,\ldots, n, \qquad (6.20)$$

$$x_{ij} \in \{0, 1\}, \quad i = 1,\ldots, n, \ j = 1,\ldots,n. \qquad (6.21)$$

Такая задача относится к классу задач линейного дискретного (целочисленного, булевого) программирования. В силу абсолютной унимодулярности матрицы условий такой задачи (см. 5.2) многогранник условий имеет только целочисленные вершины, поэтому решение такой задачи может быть получено на основе симплекс-метода (см. 3.5). Так как задача о назначениях является частным случаем транспортной задачи, то она может быть решена и с использованием метода потенциалов (см. 6.2). Однако более эффективным в вычислительном смысле методом решения задачи о назначениях является венгерский метод, модифицированный с учетом специфики задачи.

Венгерский метод решения задачи о назначениях
(Эгервари, Кун Г., 1955)

В алгоритме [54] используется матрица стоимости назначений $C = \|c_{ij}\|$

	c_{11}	c_{12}	. . . . .	c_{1n}
	c_{21}	c_{22}	. . . . .	c_{2n}
C =	. . .	. . .	. . . . .	. . .
	. . .	. . .	. . . . .	. . .
	c_{n1}	c_{n2}	. . . . .	c_{nn}

Элементам матрицы c_{ij} соответствует стоимость соответствующего назначения i–ого кандидата на j–ю должность. В каждом столбце и каждой строке матрицы при решении задачи о назначении можно выбрать только один элемент (одно назначение, в соответствии с ограничениями 6.19 - 6.21). Тогда, если ко всем элементам некоторой строки или некоторого столбца матрицы добавить (или вычесть) одно и то же число, то решение (вариант назначения) не изменится, а изменится только стоимость решения на такое же число. Такие преобразования матрицы назначений C называются э к в и в а л е н т н ы м и преобразованиями.

Очевидно, что если из всех элементов строки c_{ij}, j = 1,…,n вычесть минимальный элемент, то в каждой строке появится как минимум один элемент, равный 0. Аналогичные преобразования можно произвести со столбцами. Тогда минимальное по стоимости назначение соответствует выбору таких нулей, по одному в каждой строке и в каждом столбце (отметим, что их может быть несколько в каждой строке и в каждом столбце матрицы C). Нули, расположенные в различных строках и столбцах матрицы C, называются н е з а в и с и м ы м и нулями. Если количество независимых нулей равно n, то задача решена.

Алгоритм состоит в последовательном проведении итераций. На каждой итерации, если число независимых нулей меньше, чем n, то производится назначение очередного независимого нуля, и, если это необходимо, то производятся эквивалентные преобразования матрицы C. Таким образом, на каждой итерации число независимых нулей увеличивается на 1.

Работа алгоритма начинается с предварительной подготовки матрицы C путем эквивалентных преобразований.

0). Начальное состояние. Производятся предварительные преобразования матрицы назначений C. Из элементов каждого столбца матрицы вычитается минимальный элемент, тогда в каждом столбце будет содержаться элемент равный 0. Соответственно, из элементов каждой строки вычитается минимальный элемент, тогда и в каждой строке будет содержаться нулевой элемент. Такая матрица обозначается как $C^0 = C^s$, s=0.

Производится выбор независимых нулей. Выбирается некоторый элемент равный 0 и помечается звездочкой (0*), строка и столбец, соответствующие данному помеченному нулю, вычеркиваются. В оставшейся части матрицы C производится аналогичная операция, и так далее до тех пор, пока в оставшейся части матрицы будут содержаться нули.

Номер итерации k = 1; s = 0.

Произвольная k-ая итерация состоит в выполнении нескольких шагов. Рассмотрим их содержание.

1). Анализ оптимальности решения. Если количество помеченных нулей (0*) равно n (размерности матрицы назначений C), то процесс окончен – найдено оптимальное назначение, которое определяется помеченными (*) независимыми нулями.

2). Анализ кандидатов. Производится анализ нулевых элементов матрицы C^s, которые могут быть кандидатами в независимые нули.

А). Производится выделение знаком (+) столбцов, соответствующих помеченным (0*) нулям (независимым нулям). Выделенные (+) столбцы будем называть з а н я т ы м и с т о л б ц а м и, аналогично, в дальнейшем выделенные знаком (+) строки будем называть з а н я т ы м и с т р о к а м и. Нули, непомеченные (*) и содержащиеся в занятых строках или столбцах будем называть з а н я т ы м и нулями, в отличие остальных нулей, которые будем называть с в о б о д н ы м и нулями.

Б). Производится анализ свободных нулей.

Если в незанятых столбцах и строках не содержится свободных нулей, то производится выполнение пункта 4 (эквивалентные преобразования матрицы).

В противном случае в незанятых столбцах матрицы C^s содержится непомеченный (*) нуль, тогда возможно два случая:

а). В строке с непомеченным нулем ***нет*** помеченного (*) нуля; тогда непомеченный нуль помечается (0′), и производится переход к пункту 3 (увеличение числа независимых нулей).

б). В строке с непомеченным нулем ***есть*** помеченный (*) нуль; тогда:
- непомеченный нуль помечается (0′);
- строка выделяется знаком (+) и является занятой строкой;
- снимается отметка с занятого столбца, соответствующего помеченному (0*) нулю (знак (+) обводится кружком ⊕ и считается незанятым);
- производится переход к пункту Б) данного шага.

Данный шаг заканчивается переходом либо к шагу 3, с последующим увеличением числа независимых нулей и переходом к шагу 1, либо к шагу 4, с преобразованием матрицы назначений C^s в C^{s+1} и последующим возвратом к выполнению шага 2.

3). Увеличение числа независимых нулей. Строится цепочка нулей, состоящая из (0′) и (0*). Начиная с последнего помеченного нуля (0′), в столбце выбирается (0*), от него по строке производится переход к (0′), далее снова по столбцу к (0*) и т. д. Процесс построения цепочки оборвется на некотором (0′) (возможно, что и на первом).

Производится замена пометки в образованной цепочке: (0*) заменяется на (0); и (0′) заменяется на (0*). Иначе, снимается отметка с (0*), а (0′) заменяется на (0*). Число независимых нулей, отмеченных как 0*, становится на 1 больше.

Снимается выделение строк и столбцов («занятость» строк и столбцов). Снимается отметка (0′).

Итерация закончена; производится переход на шаг 1; $k = k + 1$.

4). Эквивалентные преобразования матрицы. Среди элементов C^s, соответствующих незанятым строкам и столбцам, выбирается минимальный элемент, который вычитается из этих свободных элементов и добавляется к элементам, стоящих на пересечении занятых строк и столбцов. Полученная матрица C^{s+1} эквивалентна матрице C^s и содержит свободные нули; $s = s+1$.

В случае, когда задача о назначениях решается на максимум целевой функции, алгоритм отличается выполнением 0-го шага.

0). При выполнении предварительных преобразований матрицы назначений в каждом столбце выбирается максимальный элемент, из которого последовательно вычитаются все остальные элементы столбца. Затем в каждой строке выбирается минимальный элемент, который вычитается из каждого элемента данной строки.

Рассмотрим пример.

Пример.

Применим рассмотренный алгоритм для решения задачи о назначениях, если матрица стоимости назначений имеет вид

	3	5	5	2	2	
	5	3	4	3	1	
C =	4	5	6	3	3	
	2	3	7	2	1	
	7	7	8	4	5	

0). Произведем предварительные преобразования матрицы C. Из элементов каждого столбца и каждой строки вычтем минимальные элементы. В результате получим матрицу C^0.

	1	2	1	0*	1	
	3	0*	0	1	0	
C^0 =	1	1	1	0	1	
	0*	0	3	0	0	
	3	2	2	0	2	

Итерация 1.

1). Анализ оптимальности решения. Количество помеченных нулей (0*) равно 3, что меньше размерности C^0 (n=5).

2). Анализ кандидатов.

А). Производится выделение знаком (+) 1, 2, 3 столбцов.

	+	+		+		
	1	2	1	0*	1	
	3	0*	0	1	0	
C^0 =	1	1	1	0	1	
	0*	0	3	0	0	
	3	2	2	0	2	

Б). Производится анализ свободных нулей.

б). В строке 2 с непомеченным нулем есть помеченный (*) нуль; непомеченный нуль помечается (0′); строка выделяется знаком (+) и является занятой строкой; снимается отметка со 2-го занятого столбца, соответствующего помеченному (0*) нулю (знак (+) обводится кружком ⊕ и считается незанятым). Переход на начало пункта Б.

б). В строке 4 с непомеченным нулем есть помеченный (*) нуль; непомеченный нуль помечается (0′); строка выделяется знаком (+); снимается отметка с 1-го занятого столбца, соответствующего помеченному (0*) нулю. Переход на начало пункта Б.

В незанятых столбцах и строках не содержится свободных (не помеченных) нулей. Производится выполнение пункта 4.

Матрица C^0 имеет вид:

	$\oplus$	$\oplus$		+		
	1	2	1	0*	1	
	3	0*	0′	1	0	+
C^0 =	1	1	1	0	1	
	0*	0′	3	0	0	+
	3	2	2	0	2	

4). Эквивалентные преобразования матрицы. Минимальный свободный элемент равен 1, он вычитается из незанятых элементов 1,3,5 строки и добавляется к элементам, стоящим на пересечении 4-го столбца и 2, 4 строки.
В результате получим матрицу C^1

				+		
	0	1	0	0*	0	
	3	0*	0′	2	0	+
C^1 =	0	0	0	0	0	
	0*	0′	3	1	0	+
	2	1	1	0	1	

2). Анализ кандидатов.

Б). Производится анализ свободных нулей.

б). В строке 1 с непомеченным нулем есть помеченный (*) нуль.

Переход на начало пункта Б.

а). В строке 3 с непомеченным нулем нет помеченного (*) нуля; тогда непомеченный нуль помечается (0′), и производится переход к пункту 3.

				$\oplus$		
	0′	1	0	0*	0	+
	3	0*	0′	2	0	+
C^1 =	0′	0	0	0	0	
	0*	0′	3	1	0	+
	2	1	1	0	1	

3). Увеличение числа независимых нулей. Строится цепочка нулей, состоящая из (0′) и (0*), начиная с последнего помеченного (0′): 0′(3,1) - 0*(4,1) - 0′(4,2) - 0*(2,2) - 0′(2,3). Производится замена в элементах цепочки: 0*(3,1) - 0(4,1) - 0*(4,2) - 0(2,2) - 0*(2,3).

Итерация 1 закончена

Итерация 2.

1). Анализ оптимальности решения. Количество помеченных нулей (0*) равно 4 (<5).

2). Анализ кандидатов.

А). Производится выделение знаком (+) 1, 2, 3, 4 столбцов.

Б). Производится анализ свободных нулей.

б). В строке 1 с непомеченным нулем есть помеченный (*) нуль; непомеченный нуль помечается (0′); строка выделяется (+);

снимается отметка с 4 столбца; производится переход на пункт Б.

Б).

б). В строке 3 с непомеченным нулем есть помеченный (*) нуль.

Переход на начало пункта Б.

Б).

	⊕	⊕	⊕	⊕		
	0	1	0	0*	0′	+
	3	0′	0*	2	0	+
C^1 =	0*	0	0	0′	0	+
	0′	0*	3	1	0	+
	2	1	1	0′	1	

б). В строке 4 с непомеченным нулем есть помеченный (*) нуль. Переход на начало пункта Б.

Б). б). В строке 2 с непомеченным нулем есть помеченный (*) нуль. Переход на начало пункта Б.

Б). а). В строке 5 с непомеченным нулем нет помеченного (*) нуля. Переход на пункт 3.

3). Увеличение числа независимых нулей. Строится цепочка нулей, состоящая из (0′) и (0*), начиная с последнего помеченного (0′): 0′(5,4) - 0*(1,4) - 0′(1,5). Производится замена в элементах цепочки: 0*(5,4) - 0(1,4) - 0*(1,5).

Итерация 2 закончена.

Итерация 3.

1). Анализ оптимальности решения. Количество помеченных нулей (0*) равно 5, что равно размерности матрицы.

Получено оптимальное решение: $x_{15}=x_{23}=x_{31}=x_{42}=x_{54}=1$.

Оптимальное значение целевой функции равно 2 + 4 + 4 + 3 + 4 = 17.

	0	1	0	0	0*	
	3	0	0*	2	0	
C^1 =	0*	0	0	0	0	
	0	0*	3	1	0	
	2	1	1	0*	1	

Контрольные вопросы

1. В чем состоит специфика транспортной задачи по сравнению с другими задачами линейного программирования ?
2. В чем заключается преимущество метода минимального элемента при построении опорного плана транспортной задачи ?
3. В чем смысл критерия оптимальности решения в методе потенциалов ?
4. Каким образом учитываются возможности движения груза при решении транспортной задачи в сетевой постановке ?
5. За счет чего обеспечивается целочисленность решения задачи о назначениях при использовании венгерского метода ?

7. ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

7.1. Постановка задач математического программирования

Проанализируем особенности построения модели принятия решения в виде задачи математического программирования. Пусть концептуальная модель принятия решения задается следующим содержательным описанием.

Задача.

Для инженерного оборудования позиционного района отдельного командно-измерительного комплекса (ОКИК) необходимо обеспечить четыре строительные площадки материалами, которые могут быть доставлены с баз, расположенных в трех близлежащих населенных пунктах, численностью соответственно 2, 5 и 8 тысяч жителей. Требуется спланировать, каким образом следует осуществить перевозки материалов, с тем чтобы минимизировать расход бензина, необходимого для работы автомобильного транспорта (грузоподъемность одной машины 5 т).

Потребность в материалах на строительных площадках (СП) составляет соответственно 25, 30, 15, 20 тонн. На складах населенных пунктов (НП) имеется соответственно материалов в количестве 62, 45, 37 тонн. Расход бензина (в литрах), необходимого для доставки материалов одной машиной, задается таблицей

	$СП_1$	$СП_2$	$СП_3$	$СП_4$
$НП_1$	19	12	18	21
$НП_2$	10	16	13	14
$НП_3$	29	14	16	15

Анализ ситуации принятия решения.

Решение заключается в планировании перевозок материалов из населенных пунктов на строительные площадки ОКИК. На данное решение влияют потребности в материалах на строительных площадках и имеющиеся ресурсы на складах населенных пунктов, и не оказывает влияния численность жителей. Общее количество материалов на всех базах достаточно для обеспечения потребностей всех строительных площадок.

Решение должно содержать данные о том, сколько машин с грузом материалов следует отправить из какого-либо населенного пункта на некоторую строительную площадку ОКИК. Таким образом, решение (план перевозок) можно описать совокупностью переменных $x_{ij} \geq 0$, характеризующих количество машин, необходимых для перевозки груза из i-го населенного пункта на j-ую строительную площадку.

<u>Разработка модели принятия решения</u>.

Модель принятия решения в общем случае описывается конструкцией вида (Δ,f), где Δ - множество допустимых решений, f - правило выбора решения. Решение x однозначно описывается вектором $x = \|x_{ij}\|$, $\forall x_{ij} \geq 0$, i = 1,2,3; j = 1,2,3,4 компоненты которого характеризуют количество машин, необходимых для перевозки материалов из i-го пункта на j-ую площадку. Допустимое решение $x \in \Delta$ должно быть практически реализуемо, для чего оно должно удовлетворять условиям поставленной задачи. Следовательно, прежде всего, необходимо провести формализацию условий задачи в виде ограничений некоторого вида.

На каждую строительную площадку следует доставить необходимое количество материалов. Так как суммарное количество материалов на всех базах больше, чем это необходимо для проведения работ на строительных площадках, то потребности в материалах могут быть полностью удовлетворены. Эти условия можно формально описать совокупностью ограничений

$$\sum_{i=1}^{3} 5x_{i1} = 25,\ \sum_{i=1}^{3} 5x_{i2} = 30, \sum_{i=1}^{3} 5x_{i3} = 15,\ \sum_{i=1}^{3} 5x_{i4} = 20\,.$$

С каждой базы может быть выделено материалов не больше, чем их имеется там в наличии. Эти условия могут быть формально описаны совокупностью неравенств

$$\sum_{j=1}^{4} 5x_{1j} \leq 62,\ \sum_{j=1}^{4} 5x_{2j} \leq\ 45, \sum_{j=1}^{4} 5x_{3j} \leq 37.$$

В задаче следует найти решение, минимизирующее суммарный расход бензина, необходимого для доставки грузов. Следовательно, качество решения можно описать следующей целевой функцией

$$L(x) = \sum_{i=1}^{3}\sum_{j=1}^{4} f_{ij}\, x_{ij},$$

где f_{ij} - количество бензина, необходимого для передвижения одной машины от i-го населенного пункта до j-ой строительной площадки (см. таблицу). Наилучшим будет решение с наименьшими значениями целевой функции L(x). Тогда правило выбора будет иметь вид

$$L(x) = \sum_{i=1}^{3}\sum_{j=1}^{4} f_{ij}\, x_{ij} \to \min .$$

В целом задача выбора наилучшего решения x^* описывается следующей конструкцией

$$x^* = \arg \max_{x \in \Delta} L(x),$$

где Δ - множество решений, удовлетворяющих поставленным ограничениям.

Анализ задачи показывает, что целевая функция L(x) является линейной функцией; множество допустимых альтернатив (решений) задается линейными ограничениями. Следовательно, разработанная модель принятия решения является задачей линейного программирования и решение может быть найдено на основе использования соответствующих алгоритмов (симплекс-метода, например).

<u>Выбор решения</u>.

Нахождение наилучшего решения основывается на использовании алгоритма симплекс-метода, для чего задача должна быть представлена в каноническом виде, то есть неравенства на основе ввода дополнительных переменных $x_{д1}$, $x_{д2}$, $x_{д3}$ сводятся к виду равенств. Тогда задача принятия решения в каноническом виде

$$L(x) = \sum_{i=1}^{3}\sum_{j=1}^{4} f_{ij}\, x_{ij} \to \min .$$

при ограничениях

$$5x_{11} + 5x_{21} + 5x_{31} = 25$$
$$5x_{12} + 5x_{22} + 5x_{32} = 30$$
$$5x_{13} + 5x_{23} + 5x_{33} = 15$$

$5x_{14} + 5x_{24} + 5x_{34} = 20$
$5x_{11} + 5x_{12} + 5x_{13} + 5x_{14} + x_{д1} = 62$
$5x_{21} + 5x_{22} + 5x_{23} + 5x_{24} + x_{д2} = 45$
$5x_{31} + 5x_{32} + 5x_{33} + 5x_{34} + x_{д3} = 37$
$x_{ij} \geq 0$, $i=1,2,3$, $j = 1,2,3,4$.

Решение, которое ищется в данной задаче, однозначно описывается вектором x, имеющим 15 неотрицательных компонент $x = \| x_{11}\ x_{12}\ x_{13}\ x_{14}\ x_{21}\ x_{22}\ x_{23}\ x_{24}\ x_{31}\ x_{32}\ x_{33}\ x_{34}\ x_{д1}\ x_{д2}\ x_{д3} \|^{т}$. Тогда задача может быть представлена в векторно матричной форме

$$L = f^{т}x \rightarrow \min$$
$$A\,x = b,$$
$$x \geq 0,$$

или

$$\| 19\ 12\ 18\ 21\ 10\ 16\ 13\ 14\ 29\ 14\ 16\ 15\ 0\ 0\ 0 \|\ x \rightarrow \min$$

	5	0	0	0	5	0	0	0	5	0	0	0	0	0	0		25
	0	5	0	0	0	5	0	0	0	5	0	0	0	0	0		30
	0	0	5	0	0	0	5	0	0	0	5	0	0	0	0		15
A =	0	0	0	5	0	0	0	5	0	0	0	5	0	0	0	x =	20
	5	5	5	5	0	0	0	0	0	0	0	0	1	0	0		62
	0	0	0	0	5	5	5	5	0	0	0	0	0	1	0		45
	0	0	0	0	0	0	0	0	5	5	5	5	0	0	1		37

Решение данной задачи на основе симплекс-метода позволяет получить следующие результаты:

- минимальное значение целевой функции, равное 220.0;
- решение прямой задачи

$$x^{*} = \| 0\ 6\ 0\ 0\ 5\ 0\ 3\ 1\ 0\ 0\ 0\ 3\ 32\ 0\ 22 \|^{т};$$

- решение двойственной задачи

$$\mu^{*} = \| 2.2\ \ 2.4\ \ 2.8\ \ 3.0\ \ 0\ -0.2\ \ 0 \|^{т}.$$

Анализ результатов решения.

В результате решения поставленной задачи получен план перевозок материалов из населенных пунктов на строительные площадки ОКИК, при этом минимальный суммарный расход бензина составляет 220 литров.

Анализируя результаты решения прямой задачи линейного программирования ($x^* = \| 0\ 6\ 0\ 0\ 5\ 0\ 3\ 1\ 0\ 0\ 0\ 3\ 32\ 0\ 22 \|^T$), можно заключить, что

- на первую строительную площадку целесообразно отправить 5 машин с базы, расположенной во 2 населенном пункте;
- на вторую строительную площадку целесообразно отправить 6 машин с базы, расположенной в 1 населенном пункте;
- на третью строительную площадку целесообразно отправить 3 машины с базы, расположенной во 2 населенном пункте;
- на четвертую строительную площадку целесообразно отправить 1 машину с базы, расположенной во 2 населенном пункте, и 3 машины с базы, расположенной в 3 населенном пункте;

При этом со второй базы будут вывезены все материалы, т.к. $x_{д2} = 0$; на первой базе останется 32 и на третьей 22 тонны материала.

Анализ результатов решения двойственной задачи показывает, что ($\mu^* = \| 2.2\ 2.4\ 2.8\ 3.0\ 0\ -0.2\ 0 \|^T$):

- оптимальное значение целевой функции прямой и двойственной задачи равны $c^T x^* = b^T \mu^* = 220$;
- если потребности строительных площадок в материалах будут увеличиваться, то расход бензина будет расти, причем более всего он будет расти, если увеличиваются потребности $СП_4$ (т.к. $\mu_4 = 3.0$), и менее всего, если увеличиваются потребности $СП_1$ (т.к. $\mu_1 = 2.2$);
- увеличение запасов материалов на базах пунктов $НП_1$ и $НП_3$ не влияет на расход бензина ($\mu_1 = 0$, $\mu_3 = 0$), в то же время увеличение количества материалов на базе $НП_2$ позволяет снизить расход бензина, необходимого на перевозку ($\mu_2 = -0.2$).

7.2. Линейное программирование

Рассмотрим задачу линейного программирования, поставленную в следующем виде

$$L = c^T x \to \max$$
$$A x \le b$$
$$x \ge 0$$

Здесь A - матрица условий размерности $m \times n$;

c^T - вектор коэффициентов целевой функции размерности $1 \times n$;

b - вектор ограничений размерности $m \times 1$;

x - решение.

7.2.1. Графо-аналитический способ решения задач (см.п.3.3)

Графо-аналитический способ решения задач обладает достаточной наглядностью, позволяет уяснить основные особенности задач линейного программирования, с его помощью можно решать задачи линейного программирования, когда m=2 или n=2.

Рассмотрим особенности использования метода на примере следующей задачи

<u>Пример 7.1.</u>

$$2x_1 + 14x_2 + 22x_3 + 4x_4 \to \min$$
$$3x_1 + 1x_2 - 4x_3 - 2x_4 \le 1$$
$$1x_1 + 3x_2 + 1x_3 - 3x_4 \ge 1$$
$$x_j \ge 0,\ j = 1,2,3,4.$$

Приведем задачу к стандартному виду

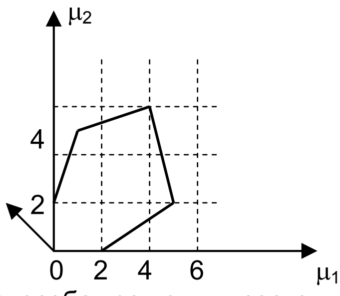
$$-2x_1 - 14x_2 - 22x_3 - 4x_4 \to \max$$
$$3x_1 + 1x_2 - 4x_3 - 2x_4 \le 1$$
$$-1x_1 - 3x_2 - 1x_3 + 3x_4 \le -1$$
$$x_j \ge 0,\ j = 1,2,3,4.$$

Для данной задачи запишем двойственную задачу

$$1\mu_1 - 1\mu_2 \to \min$$
$$3\mu_1 - 1\mu_2 \ge. -2$$
$$1\mu_1 - 3\mu_2 \ge -14$$
$$-4\mu_1 - 1\mu_2 \ge -22$$
$$-2\mu_1 + 3\mu_2 \ge -4$$
$$\mu_1, \mu_2 \ge 0$$

Избавляемся от "-" в правой части.

$$-1\mu_1 + 1\mu_2 \to \max$$
$$-3\mu_1 + 1\mu_2 \le 2$$
$$-1\mu_1 + 3\mu_2 \le 14$$
$$4\mu_1 + 1\mu_2 \le 22$$
$$2\mu_1 - 3\mu_2 \le 4,\quad \mu_1, \mu_2 \ge 0 .$$



Суть графо-аналитического способа решения состоит в том, что градиент, затем строится линия уровня $L(\mu) = -1\mu_1 + 1\mu_2 = A$ (в общем случае это гиперплоскость), которая перпендикулярна градиенту. Все решения, находящиеся на этой линии, имеют одинаковое значение целевой функции равное A. Далее эта линия перемещается параллельно самой себе в направлении возрастания градиента до тех пор, пока линия уровня не достигнет границы допустимой области. Решения, которые соответствуют пересечению линии уровня и границы допустимой области, являются оптимальными.

Для рассматриваемой задачи оптимальное решение соответствует точке (1,5), L (1,5) = -4.

Проведем анализ полученного решения двойственной задачи.

♦ 3-е и 4-е уравнения выполняются как строгие неравенства, следовательно, в соответствии с условиями дополняющей нежесткости (см. ф.(2.6) и п.3.1) $x_3 = x_4 = 0$.

♦ 1-е и 2-е уравнения выполняются как равенства, следовательно, в соответствии с условиями дополняющей нежесткости (см. ф.(2.6) и п.3.1) $x_1 > 0$, $x_2 > 0$.

С учетом проведенного анализа, в результате которого было установлено, что $x_3 = x_4 = 0$ исходную задачу можно переписать как $L = -2x_1 - 14x_2$

$$3x_1 + 1x_2 = 1$$
$$1x_1 + 3x_2 = 1$$

Откуда $x_1 = 0.25$; $x_2 = 0.25$; $L = -4$.

Так как в процессе приведения исходной задачи к стандартному виду была проведена замена целевой функции, то окончательное решение имеет вид $x^* = (0.25\ 0.25\ \ 0\ \ 0)$; $L^* = 4$.

Замечание 1.

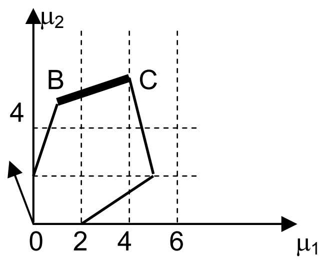
Если решение исходной задачи ***вырожденное***, то оптимальное решение двойственной задачи достигается в ***нескольких крайних точках*** (множество оптимальных решений). Действительно, пусть исходная задача имеет вид

<u>Пример 7.2.</u>

$$2x_1 + 14x_2 + 22x_3 + 4x_4 \to \min$$
$$3x_1 + 1x_2 - 4x_3 - 2x_4 \le 1$$
$$1x_1 + 3x_2 + 1x_3 - 3x_4 \ge 3 \quad x_j \ge 0,\ j = 1,2,3,4.$$

Тогда двойственная задача

$$-1\mu_1 + 3\mu_2 \to \max$$
$$-3\mu_1 + 1\mu_2 \le 2$$
$$-1\mu_1 + 3\mu_2 \le 14$$
$$4\mu_1 + 1\mu_2 \le 22$$
$$2\mu_1 - 3\mu_2 \le 4$$
$$\mu_1,\ \mu_2 \ge 0$$



Оптимальное решение двойственной задачи достигается в двух крайних точках B и C (следовательно, в задаче бесконечное множество оптимальных решений – отрезок BC). В этом случае решение исходной задачи является вырожденным

$x^* = (0\ 1\ 0\ 0)$; $L^* = 14$.

Замечание 2.

Если целевая функция исходной задачи может ***неограниченно возрастать*** (убывать), т.е. оптимального решения не существует, то множество допустимых решений двойственной задачи ***пусто*** ($\Delta = \varnothing$).

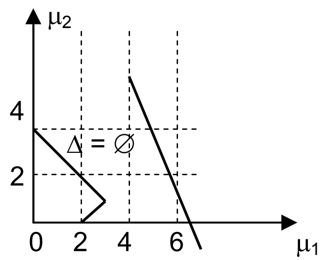
<u>Пример 7.3.</u>

Рассмотрим задачу

$$
\begin{aligned}
2x_1 + 4x_2 - 14x_3 &\to \min\\
1x_1 + 1x_2 - 2x_3 &\ge 1\\
-1x_1 + 1x_2 - 1x_3 &\ge 2\\
x_j \ge 0,\ j &= 1,2,3,4.
\end{aligned}
$$

Задача двойственная к данной:

$$
\begin{aligned}
1\mu_1 + 2\mu_2 &\to \max\\
1\mu_1 - 1\mu_2 &\le 2\\
1\mu_1 + 1\mu_2 &\le 4\\
2\mu_1 + 1\mu_2 &\ge 14\\
\mu_1, \mu_2 &\ge 0
\end{aligned}
$$



Множество допустимых решений двойственной задачи пусто. При решении прямой задачи, если $x_1 = 0$; а переменные x_2, x_3 изменяются в пропорции $3x_2 = x_3$, то все такие решения при любых x_2, x_3 являются допустимыми, в тоже время целевая функция неограниченно убывает.

Замечания 1 и 2 справедливы, если на место прямой задачи поставить двойственную задачу, а на место двойственной - прямую.

7.2.2. Симплекс-метод решения задач

<u>Каноническая форма задачи</u>

Симплекс-метод является универсальным методом решения задач линейного программирования. Для его использования задача должна быть сведена к канонической форме (ограничения должны быть представлены в виде равенств). В канонической форме может быть представлена любая задача линейного программирования.

Действительно, пусть имеется s-е ограничение вида

$\sum_{j=1}^{n} a_{sj}x_j \le b_s$, тогда в канонической форме $\sum_{j=1}^{n} a_{sj}x_j + x_{n+1} = b_s$

Соответственно, если

$\sum_{j=1}^{n} a_{kj}x_j \ge b_k$, тогда в канонической форме. $\sum_{j=1}^{n} a_{kj}x_j - x_{n+1} = b_k$

Если переменные задачи могут принимать отрицательные значения, т.е имеет место задача вида $c^T x \to \max$

$A\,x \le b$,

то ее следует преобразовать к виду:

$c^T y - c^T z \to \max$

$Ay - A\,z \le b$

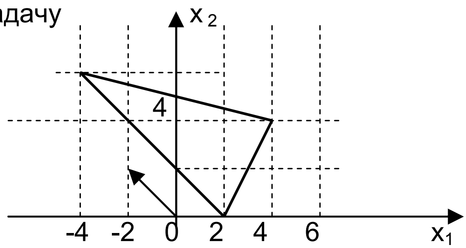
$y, z \ge 0$.

Если в результате решения задачи $y_j > 0$, то $x_j = y_j > 0$; если $z_k > 0$, то $x_{k-n} = -z_k < 0$.

<u>Пример 7.4.</u>

Рассмотрим следующую задачу

$$
\begin{aligned}
-1x_1 + 1x_2 &\to \max \\
1x_1 + 1x_2 &\ge 2 \\
2x_1 - 1x_2 &\le 4 \\
1x_1 + 4x_2 &\le 20 \\
x_1, x_2 &\in (-\infty, +\infty),
\end{aligned}
$$



По условиям задачи переменные могут принимать о т р и ц а т е л ь н ы е значения. Задачу следует переписать в виде

$$
\begin{aligned}
-1y_1 + 1y_2 + 1z_1 - 1z_2 &\to \max \\
1y_1 + 1y_2 - 1z_1 - 1z_2 &\ge 2 \\
2y_1 - 1y_2 - 2z_1 + 1z_2 &\le 4 \\
1y_1 + 4y_2 - 1z_1 - 4z_2 &\le 20, \\
y_1, y_2, z_1, z_2 &\ge 0.
\end{aligned}
$$

Представим данную задачу в каноническом виде, для этого сведем ограничения вида неравенств к равенствам, тогда

$$
\begin{aligned}
-1x_1 + 1x_2 + 1x_3 - 1x_4 &\to \max \\
1x_1 + 1x_2 - 1x_3 - 1x_4 - 1x_5 &= 2
\end{aligned}
$$

$$2x_1 - 1x_2 - 2x_3 + 1x_4 \qquad +1x_6 \quad = 4$$
$$1x_1 + 4x_2 - 1x_3 - 4x_4 \qquad\qquad +1x_7 = 20,$$
$$x_1,\dots,x_7 \ge 0.$$

В результате решения данной задачи (см. Пример.7.9) получим, что оптимальное значение целевой функции равно 10, а оптимальное решение имеет вид $x^* = (\ 0\ 6\ 4\ 0\ 0\ 18\ 0\ )^т$, т.е. x_1=-4; x_2=6.

<u>Алгоритм симплекс-метода</u>

(см. пп. 3.4, 3.5).

Рассмотрим особенности проведения итераций симплекс-метода.

<u>Пример 7.5.</u>

$$2x_1 + 1x_2 \to \max$$
$$-1x_1 + 1x_2 \le 3$$
$$2x_1 - 1x_2 \le 4$$
$$1x_1 + 2x_2 \le 12,$$
$$x_1, x_2 \ge 0.$$

В канонической форме задача имеет вид

$$2x_1 + 1x_2 \to \max$$
$$-1x_1 + 1x_2 + 1x_3 \qquad = 3$$
$$2x_1 - 1x_2 \qquad + 1x_4 \quad = 4$$
$$1x_1 + 2x_2 \qquad\qquad + 1x_5 = 12, \quad x_1,x_2,x_3,x_4,x_5 \ge 0.$$

В векторно-матричной форме

$$\| \ 2\ \ 1\ \ 0\ \ 0\ \ 0 \ \| \ x \to \max$$

	-1	1	1	0	0		3
	2	-1	0	1	0	x =	4
	1	2	5	0	1		12

Поскольку задача поставлена в стандартной форме, то исходное опорное решение очевидно: $x = (0\ 0\ 3\ 4\ 12)^т$. Исходная симплекс-таблица имеет вид

0	0.0	0.0	0.0	0.0		
3	3.0	1.0	0.0	0.0		
4	4.0	0.0	1.0	0.0		
5	12.0	0.0	0.0	1.0		

Итерация 1:

1. Анализ оптимальности текущего решения; нахождение вектора, вводимого в базис.

Рассчитываются оценки (3.13)

$\delta_k = \min (\pi^T a_j - c_j),\ j = 1,...,5.$

$\delta_k = \min \{-2\ -1\ 0\ 0\ 0\} = -2;\ \ k = 1.$

Т.к. $\delta_k < 0$, k-ый вектор условий следует ввести в базис.

2. Установление неразрешимости задачи; нахождение вектора, выводимого из базиса.

Вектор a_k приводится к текущему базису - находится вектор $æ_k = S^{-1} a_k$. Т.к. $\exists\ æ_{ki} > 0$, i=1,2,3, то задача разрешима и производится вычисление оценки θ_r по формуле (3.15)

$$\theta_r = \min_{æ_{ki}>0} \frac{x_{si}}{æ_{ki}}$$

$\theta_r = \min \{ 2\ \ 12 \} = 2;\ r = 2.$

Приращение целевой функции (см.ф.3.12) $\Delta L = - \delta_k \theta_r = 4.$

3. Преобразование базиса.

Исходная симплекс-таблица имеет вид

0	0.0	0.0	0.0	0.0	-2.0	-2.0
3	3.0	1.0	0.0	0.0	-1.0	-1.0
4	4.0	0.0	1.0	0.0	2.0	2.0
5	12.0	0.0	0.0	1.0	1.0	1.0

Преобразуем ее по рекуррентным формулам (3.17),(3.18)

1	4.0	0.0	1.0	0.0	-2.0	-2.0
3	5.0	1.0	0.5	0.0	0.5	1.0
1	2.0	0.0	0.5	0.0	-0.5	-1.0
5	10.0	0.0	-0.5	1.0	2.5	2.0

Итерация 2:

1. $\delta_k = -2;\ \ k = 2.$
2. $\theta_r = 4;\ \ \ r = 3.$
3.

2	12.0	0.0	0.6	0.8
3	3.0	1.0	0.6	-0.2
1	4.0	0.0	0.4	0.2
2	4.0	0.0	-0.2	0.4

Итерация 3:

1. $\delta_k = 0$, следовательно, текущее решение оптимально.

Таким образом, оптимальное решение поставленной задачи линейного программирования $x^* = (4\ 4\ 3\ 0\ 0)^т$; $L^* = 12$.

Решение двойственной задачи (0 0.6 0.8).

Неограниченность целевой функции

(см.п.3.4.3).

Ситуации, когда целевая функция неограниченно возрастает на множестве допустимых решений, выявляются на втором шаге алгоритма симплекс-метода, а именно при анализе вектора, выводимого из базиса, так, если все компоненты $æ_{ki} \le 0$, то имеет место такая ситуация.

Пример 7.6.

Рассмотрим задачу вида

$$\left\| 2\ \ 3\ \ 5\ \ 0\ \ 0 \right\| x \to \max$$

$$\left\| \begin{array}{ccccc} -1 & -2 & 1 & 1 & 0 \\ 3 & -1 & -1 & 0 & 1 \end{array} \right\| x = \left\| \begin{array}{c} 2 \\ 1 \end{array} \right\|$$

$x \ge 0$.

Решим данную задачу с использованием симплекс-метода.

Итерация 1: $\delta_k = -5$; $k = 3$; $\theta_r = 2$; $r = 1$.

1	0.0	0.0	0.0	-5.0	-5.0
4	2.0	1.0	0.0	1.0	1.0
5	1.0	0.0	1.0	-1.0	-1.0

Итерация 2: $\delta_k = -13$; $k = 2$; $\theta_r = \ldots\ldots$

1	10.0	5.0	0.0	-13.0	-13.0
3	2.0	1.0	0.0	-2.0	-2.0
5	3.0	1.0	1.0	-3.0	-1.0

Все компоненты $æ_{ki} < 0$, следовательно, целевая функция может неограниченно возрастать. Действительно, второй переменной можно придавать сколь угодно большие значения, при этом ограничения задачи будут выполняться, а целевая функция неограниченно возрастает.

7.2.3. Способы построения исходного решения

(см.п.3.6)

"Очевидное" исходное опорное решение

Задачи линейного программирования в стандартной постановке имеют очевидное исходное опорное решение (см. примеры 7.5, 7.6).

Вспомогательная задача

Для нахождения исходного опорного решения может быть использована вспомогательная задача, которая ставится в виде

$$-1^T x_d \to \max,$$
$$A\,x + E\,x_d = b,$$
$$x \geq 0,\ x_d \geq 0.$$

Рассмотрим пример

Пример 7.7.

$$\begin{aligned}
-1x_1 + 1x_2 &\to \max\\
1x_1 + 1x_2 &\geq 2\\
2x_1 - 1x_2 &\leq 4\\
1x_1 + 4x_2 &= 20\\
x_1, x_2 &\geq 0.
\end{aligned}$$

Сведем задачу к каноническому виду

$$\begin{aligned}
-1x_1 + 1x_2 &\to \max\\
1x_1 + 1x_2 - 1x_3 &= 2\\
2x_1 - 1x_2 + 1x_4 &= 4\\
1x_1 + 4x_2 &= 20,\\
x_1,\dots,x_4 &\geq 0.
\end{aligned}$$

Вспомогательная задача, позволяющая найти исходное опорное решение данной задачи, имеет вид

$$\begin{aligned}
- x_5 - x_6 - x_7 &\to \max\\
1x_1 + 1x_2 - 1x_3 + x_5 &= 2\\
2x_1 - 1x_2 + 1x_4 + x_6 &= 4\\
1x_1 + 4x_2 + x_7 &= 20,\\
x_1,\dots,x_7 &\geq 0.
\end{aligned}$$

Решим ее с использованием симплекс-метода. Исходное опорное решение очевидно: $x = (0\ 0\ 0\ 0\ 2\ 4\ 20)^T$.

Итерация 1: $\delta_k = -4$; $k = 1$; $\theta_r = 2$; $r = 1$.

1	-26.0	-1.0	-1.0	-1.0	-4.0	-4.0
5	2.0	1.0	0.0	0.0	1.0	1.0
6	4.0	0.0	1.0	0.0	2.0	2.0
7	20.0	0.0	0.0	1.0	1.0	1.0

Итерация 2: $\delta_k = -3$; $k = 3$; $\theta_r = 0$; $r = 2$.

2	-18.0	3.0	-1.0	-1.0	-4.0	-4.0
1	2.0	1.0	0.0	0.0	-1.0	-1.0
6	0.0	-2.0	1.0	0.0	2.0	0.0
7	18.0	-1.0	0.0	1.0	1.0	0.0

Итерация 3: $\delta_k = -4.5$; $k = 2$; $\theta_r = 4$; $r = 3$.

3	-18.0	0.0	0.5	-1.0	-4.5	-4.5
1	2.0	1.0	0.5	0.0	-0.5	1.0
3	0.0	-1.0	0.5	0.0	-1.5	-1.0
7	18.0	0.0	-0.5	1.0	4.5	4.0

Итерация 4: $\delta_k = 0$; - решение оптимально.

4	0.0	0.0	0.0	0.0
1	4.0	0.0	0.44	0.11
3	6.0	-1.0	0.33	0.33
2	4.0	0.0	-0.11	0.22

Значение целевой функции равно 0, следовательно, оптимальное решение вспомогательной задачи является опорным решением основной задачи, и оно может быть взято в качестве исходного опорного решения x_s.

Далее, для перехода к решению основной задачи необходимо произвести замену 0-ой строки таблицы по правилу

$L(x_s) = c^T_s x_s$; $\pi^T = c^T_s S^{-1}$.

Тогда симплекс таблица будет иметь вид:

Итерация 5: $\delta_k = -0.56$; $k = 4$; $\theta_r = 9$; $r = 1$.

5	0.0	0.0	-0.56	0.11	-0.56	-0.56
1	4.0	1.0	0.44	0.11	0.44	1.0
3	6.0	-1.0	0.33	0.33	0.33	-1.0
2	4.0	0.0	-0.11	0.22	-0.11	4.0

Итерация 6: $\delta_k = 0$; - решение оптимально.

6	5.0	0.0	0.0	0.25
4	9.0	0.0	1.0	0.25
3	3.0	-1.0	0.0	0.25
2	5.0	0.0	0.0	0.25

В ходе решения вспомогательной задачи выявляется факт отсутствия в основной задаче допустимых решений. Действительно, если в оптимальном решении вспомогательной задачи присутствуют дополнительные переменные, то, следовательно, не существует ни одного набора переменных основной задачи, позволяющего сформировать базис. Рассмотрим задачу, приведенную в примере 7.3:

Пример 7.8.

$$
\begin{aligned}
&1x_1 + 2x_2 \to \max \\
&1x_1 - 1x_2 \le 2 \\
&1x_1 + 1x_2 \le 4 \\
&2x_1 + 1x_2 \ge 14 \\
&x_1, x_2 \ge 0
\end{aligned}
$$

Приведем задачу к канонической форме

$$
\begin{aligned}
&1x_1 + 2x_2 \to \max \\
&1x_1 - 1x_2 + x_3 = 2 \\
&1x_1 + 1x_2 + x_4 = 4 \\
&2x_1 + 1x_2 - x_5 = 14 \\
&x_1, \dots, x_5 \ge 0.
\end{aligned}
$$

Решим вспомогательную задачу с использованием симплекс-метода:

Итерация 1: $\delta_k = -4$; $k = 1$; $\theta_r = 2$; $r = 1$.

1	-20.0	-1.0	-1.0	-1.0	-4.0	-4.0
6	2.0	1.0	0.0	0.0	1.0	1.0
7	4.0	0.0	1.0	0.0	1.0	1.0
8	14.0	0.0	0.0	1.0	2.0	2.0

Итерация 2: $\delta_k = -5$; $k = 2$; $\theta_r = 1$; $r = 2$.

2	-12.0	3.0	-1.0	-1.0	-5.0	-5.0
1	2.0	1.0	0.0	0.0	-1.0	-1.0
7	2.0	-1.0	1.0	0.0	2.0	1.0
8	10.0	-2.0	0.0	1.0	3.0	1.0

Итерация 3: $\delta_k = 0$; - решение оптимально.

3	-7.0	0.5	1.5	-1.0		
1	3.0	0.5	0.5	0.0		
2	1.0	-0.5	0.5	0.0		

8	7.0	-0.5	-1.5	1.0		

Вспомогательная задача решена. Значение целевой функции меньше 0, следовательно, множество решений исходной задачи пусто. И это, действительно, так - см. пример 7.3.

<u>М-метод решения задачи</u>

М-метод представляет собой способ одновременного решения задачи построения исходного опорного решения и основной задачи линейного программирования. В этом случае задача имеет вид

$$c^{т} x - M^{т} x_d \to \max,$$
$$A x + E x_d = b,$$
$$x \geq 0,\ x_d \geq 0,$$

здесь вектор x_d является исходным опорным решением, а М - достаточно большие числа.

Решим задачу из примера 7.4 с использованием М-метода.

<u>Пример 7.9.</u>

$$\begin{array}{l} -1x_1 + 1x_2 + 1x_3 - 1x_4 \to \max \\ \ 1x_1 + 1x_2 - 1x_3 - 1x_4 - 1x_5 = 2 \\ \ 2x_1 - 1x_2 - 2x_3 + 1x_4 + 1x_6 = 4 \\ \ 1x_1 + 4x_2 - 1x_3 - 4x_4 + 1x_7 = 20 \\ \ x_1,\dots,x_7 \geq 0. \end{array}$$

Пусть М = 10, тогда задачу можно переписать

$$\begin{array}{l} -1x_1 + 1x_2 + 1x_3 - 1x_4 -10x_8 -10x_9 -10x_{10} \to \max \\ \ 1x_1 + 1x_2 - 1x_3 - 1x_4 - 1x_5 + 1x_8 = 2 \\ \ 2x_1 - 1x_2 - 2x_3 + 1x_4 + 1x_6 + 1x_9 = 4 \\ \ 1x_1 + 4x_2 - 1x_3 - 4x_4 + 1x_7 + 1x_{10} = 20 \\ \ x_1,\dots,x_{10} \geq 0. \end{array}$$

Используем для решения последней задачи симплекс-метод. Исходное опорное решение очевидно.

Итерация 1: $\delta_k = -41$; $k = 2$; $\theta_r = 2$; $r = 1$.

1	-260	-10.0	-10.0	-10.0	-41.0	-41.0
8	2.0	1.0	0.0	0.0	1.0	1.0

9	4.0	0.0	1.0	0.0	-1.0	-1.0
10	20.0	0.0	0.0	1.0	4.0	4.0

Итерация 2: $\delta_k = -31$; k = 5; $\theta_r = 3$; r = 3.

2	-178	31.0	-10.0	-10.0	-31.0	-31.0
2	2.0	1.0	0.0	0.0	-1.0	-1.0
9	6.0	1.0	1.0	0.0	-1.0	0.0
10	12.0	-4.0	0.0	1.0	4.0	0.0

Итерация 3: $\delta_k = -21.25$; k = 1; $\theta_r = 4$; r = 2.

3	-85	0.0	-10.0	-2.25	-21.25	-21.25
2	5.0	0.0	0.0	0.25	0.25	1.0
9	9.0	0.0	1.0	0.25	2.25	2.0
5	3.0	-1.0	0.0	0.25	-0.75	1.0

Итерация 4: $\delta_k = -0.55$; k = 6; $\theta_r = 9$; r = 2.

4	0.0	0.0	-0.55	0.11	-0.55	-0.55
2	4.0	0.0	-0.11	0.22	-0.11	0.0
1	4.0	0.0	0.44	0.11	0.44	1.0
5	6.0	-1.0	0.33	0.33	0.33	0.0

Итерация 5: $\delta_k = -1.25$; k = 3; $\theta_r = 4$; r = 3.

5	5.0	0.0	0.0	0.25	-1.25	-1.25
2	5.0	0.0	0.0	0.25	-0.25	-1.0
6	9.0	0.0	1.0	0.25	-2.25	-2.0
5	3.0	-1.0	0.0	0.25	0.75	-1.0

Итерация 6: $\delta_k = 0$; - решение оптимально.

6	10.0	-1.66	0.00	0.66		
2	6.0	-0.33	0.00	0.33		
6	18.0	-3.00	1.00	1.00		
3	4.0	-1.33	0.00	0.33		

В результате решения данной задачи получим, что оптимальное значение целевой функции равно 10, а оптимальное решение имеет вид $x^* = (0\ 6\ 4\ 0\ 0\ 18)^т$.

7.2.4. Вырожденные задачи. Зацикливание

(см. п.3.7)

В вырожденных задачах линейного программирования нарушается взаимно однозначное соответствие между опорными решениями и базисами. Одному и тому же вырожденному опорному решению может соответствовать несколько базисов (соответственно, и вершин многогранника решений). Такая ситуация может привести к зацикливанию симплекс-метода - периодическому возвращению через несколько итераций к одному и тому же базису.

Пример 7.10.

Рассмотрим задачу

$$
\begin{aligned}
&0.3x_1 - 5x_2 + 0.1x_3 - 6x_4 \to \max \\
&0.1x_1 - 2x_2 - 0.2x_3 + 9x_4 \le 0 \\
&0.2x_1 - 3x_2 - 0.1x_3 + 3x_4 \le 0 \\
&\qquad\qquad\quad 5.0x_3 \qquad\quad\ \le 1 \\
&x_1, x_2, x_3, x_4 \ge 0.
\end{aligned}
$$

Задача поставлена в стандартной форме, следовательно, она имеет очевидное исходное опорное решение. Используем симплекс-метод:

Итерация 1; $J_s = \{ 5\ 6\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 2; $J_s = \{ 1\ 6\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 3; $J_s = \{ 1\ 2\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 4; $J_s = \{ 3\ 2\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 5; $J_s = \{ 3\ 4\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 6; $J_s = \{ 5\ 4\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 7; $J_s = \{ 5\ 6\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 8; $J_s = \{ 1\ 6\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 9; $J_s = \{ 1\ 2\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 10; $J_s = \{ 3\ 2\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 11; $J_s = \{ 3\ 4\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$
Итерация 12; $J_s = \{ 5\ 4\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$

и так далее,..... Одному и тому же вырожденному опорному решению соответствуют различные базисы. На 7-ой итерации алгоритм возвратился к исходному опорному базису, и на последующих итерациях базисы повторяются. Произошло зацикливание алгоритма.

Использование рассмотренного в п.3.7 способа преодоления зацикливания позволяет получить оптимальное решение

Итерация 1; $J_s = \{ 5\ 6\ 7 \}$; $x_s = (0.0\ \ 0.0\ \ 1.0)^т$

Итерация 2; J_s = { 1 6 7 }; x_s = (0.0 0.0 1.0)^T
Итерация 3; J_s = { 1 2 7 }; x_s = (0.0 0.0 1.0)^T
Итерация 4; J_s = { 3 2 7 }; x_s = (0.0 0.0 1.0)^T
Итерация 5; J_s = { 3 4 7 }; x_s = (0.0 0.0 1.0)^T
Итерация 6; J_s = { 3 4 1 }; x_s = (0.200 0.004 0.040)^T
Итерация 7; J_s = { 3 5 1 }; x_s = (0.200 0.030 0.100)^T

Итак, оптимальное решение x^* = (0.1 0 0.2 0 0.03 0 0)^T; L^* = 0.05.

7.3. Нелинейное программирование

Будем рассматривать задачу нелинейного программирования в следующем виде

$f(x) \to \max$,
$\varphi_I(x) \le b_i$, i= 1,..., m,
$x = (x_1,x_2,...,x_n)^T \ge 0$,

где f(x) и все $\varphi_i(x)$, i = 1,..., m - выпуклые функции.

7.3.1. Градиентные методы

(см.п.4.3.1)

Градиентный метод представляет собой метод поиска экстремума функции при отсутствии ограничений. Рассмотрим следующую задачу: найти максимум функции

$f(x) = 16x_1 - 2x^2_1 + 12x_2 - x^2_2 \to \max$.

Градиентный метод с постоянным шагом.

Отдельная итерация алгоритма градиентного метода с постоянным шагом состоит в выполнении ряда шагов.

0). Исходное состояние. Задается начальная точка x^o, шаг h, коэффициент α, некоторое малое ε; k = 0.

1). Вычисляется $\nabla F(x^k)$.

2). Находится $x^{k+1} = x^k + h\,\nabla F(x^k)$.

3). Рассчитывается $f(x^{k+1})$; если $f(x^{k+1}) \le f(x^k)$, то производится дробление шага $h = \alpha h$ и производится переход на шаг 2.

4). Проверяется условие окончания процесса:

если $\|x^{k+1}-x^k\| \le \varepsilon$ или $\|\nabla F(x^k)\| \le \varepsilon$, то решение оптимально;
если $\|x^{k+1}-x^k\| > \varepsilon$ или $\|\nabla F(x^k)\| > \varepsilon$, то k=k+1; переход на шаг 1.

Пример 7.11.

0). Исходное состояние. Пусть начальная точка $x^o = (0,0)$; $h = 1$; $\alpha = 0.6$; $\varepsilon = 0.1$; $k=0$.

Итерация 1. $x^o = (0,0)$; $h = 1$; $k = 0$;

1). $\nabla F(x^o) = (16 - 4x_1, 12 - 2x_2) = (16, 12)$;

2). $x^1 = x^o + h\, \nabla F(x^o) = (16, 12)$;

3). $f(x^1) = -256 < f(x^o) = 0$ - следовательно, необходимо произвести дробление шага $h = 0.6\ h = 0.6$.

2). $x^1 = x^o + h\, \nabla F(x^o) = (9.6, 7.2)$

3). $f(x^1) = 3.84 > f(x^o) = 0$

4). $\| x^1 - x^o \| > \varepsilon$, следовательно, $k = k+1$.

Итерация 2. $x^1 = (9.6, 7.2)$; $h = 0.6$; $k = 1$;

1). $\nabla F(x^1) = (-22.4, -2.4)$

2). $x^2 = x^1 + h\, \nabla F(x^1) = (-3.84, 5.76)$

3). $f(x^2) = -55 < f(x^1) = 3.84$ - следовательно, необходимо произвести дробление шага $h = 0.6\ h = 0.36$.

2). $x^2 = (9.6, 7.2) + 0.36\ (-22.4, -2.4) = (1.54, 6.34)$;

3). $f(x^2) = 55.78 > f(x^1) = 3.84$;

4). $\| x^2 - x^1 \| > e$, тогда $k = k+1$.

Итерация 3. $x^2 = (1.54, 6.34)$; $h = 0.36$; $k = 2$;

1). $\nabla F(x^2) = (9.84, -0.68)$;

2). $x^3 = (1.54, 6.34) + 0.36\ (9.84, -0.68) = (5.08, 6.10)$;

3). $f(x^3) = 65.66 > f(x^2) = 55.78$

4). $\| x^3 - x^2 \| > \varepsilon$, тогда $k = k+1$.

Итерация 4. $x^3 = (5.08, 6.10)$; $h = 0.36$; $k = 3$;

1). $\nabla F(x^3) = (-4.32, -0.2)$;

2). $x^4 = (5.08, 6.10) + 0.36\ (-4.32, -0.20) = (3.52, 6.03)$;

3). $f(x^4) = 67.54 > f(x^3) = 65.66$

4). $\| x^4 - x^3 \| > \varepsilon$, тогда $k = k+1$.

Итерация 5. $x^4 = (3.52, 6.03)$; $h = 0.36$; $k = 4$;

1). $\nabla F(x^4) = (1.92, -0.06)$;

2). $x^5 = (3.52, 6.03) + 0.36\ (1.92, -0.06) = (4.21, 6.01)$;

3). $f(x^5) = 67.91 > f(x^4) = 67.54$

4). $\| x^5 - x^4 \| > \varepsilon$, тогда $k = k+1$.

Итерация 6. $x^5 = (4.21, 6.01)$; $h = 0.36$; $k = 5$;

1). $\nabla F(x^5) = (-0.84, -0.02)$;

2). $x^6 = (4.21, 6.01) + 0.36\ (-0.84, -0.02) = (3.91, 6.00)$;

3). $f(x^6) = 67.98 > f(x^5) = 67.91$

4). $\| x^6 - x^5 \| > \varepsilon$, тогда $k = k+1$.

Итерация 7. $x^6 = (3.91, 6.00)$; $h = 0.36$; $k = 6$;

1). $\nabla F(x^6) = (0.36, 0.0)$;
2). $x^7 = (3.91, 6.00) + 0.36\ (0.36, 0.0) = (4.04, 6.00)$;
3). $f(x^7) = 68.00 > f(x^6) = 67.98$
4). $\| x^7 - x^6 \| > \varepsilon$, тогда k = k+1. $f(x) = 16x_1 - 2x^2{}_1 + 12x_2 - x^2{}_2$.

Итерация 8. $x^7 = (4.04, 6.00)$; $h = 0.36$; $k = 7$;
1). $\nabla F(x^7) = (-0.16, 0)$;
2). $x^8 = (4.04, 6.00) + 0.36\ (-0.16, 0) = (3.98, 6.00)$;
3). $f(x^8) = 68.00 = f(x^7) = 68.00$
4). $\| x^8 - x^7 \| < \varepsilon$, следовательно, оптимальное решение получено.

Итак, оптимальное решение $x^* = (3.98, 6.0)^т$; $f(x^*) = 68$.

Градиентный метод наискорейшего подъема.

Решим ту же задачу методом наискорейшего подъема.

$$f(x) = 16x_1 - 2x^2{}_1 + 12x_2 - x^2{}_2 \to \max$$

Отдельная итерация алгоритма наискорейшего подъема состоит в выполнении следующих шагов.

0). Задается начальная точка x^o, некоторое малое ε; $k = 0$.
1). Вычисляется $\nabla F(x^k)$.
2). Рассчитывается $h_k = \max f(x^k + h\ \nabla f(x^k))$, $h \geq 0$.
3). Находится $x^{k+1} = x^k + h_k\ \nabla F(x^k)$.
4). Проверяется условие окончания процесса:
если $\|x^{k+1}-x^k\| \leq \varepsilon$ или $\|\nabla F(x^k)\| \leq \varepsilon$ - решение оптимально.
если $\|x^{k+1}-x^k\| > \varepsilon$ или $\|\nabla F(x^k)\| > \varepsilon$, то $k = k+1$.

Пример 7.12.

0). Исходное состояние. Начальная точка $x^o = (0,0)$; $\varepsilon=0.1$; $k=0$.

Итерация 1. $k = 0$; $x^o = (0,0)$;
1). $\nabla F(x^o) = (16 - 4x_1, 12 - 2x_2) = (16, 12)$;
2). $h_0 = \max f\ (x^o + h\ \nabla f\ (x^0)) = \max f\ (16h, 12h) =$
$= \max\ (256h - 512h^2 + 144h - 144h^2) = 0.3$
3). $x^1=x^o+h\nabla F(x^o) = (4.8, 3.6)$; $f(x^1) = 60.96$;
4). $\|x^1 - x^o\| > \varepsilon$, $k=k+1$.

Итерация 2. $k = 1$; $x^1 = (4.8, 3.6)$;
1). $\nabla F(x^1) = (16 - 4x_1, 12-2x_2) = (-3.2, 4.8)$
2). $h_1 = \max f(4.8 - 3.2h, 3.6 + 4.8h) = 0.38$
3). $x^2 = x^1 + h\ \nabla F(x^1) = (3.6, 5.4)$; $f(x^2) = 67.32$;
4). $\|x^2 - x^1\| > \varepsilon$, $k = k+1$.

Итерация 3. $k = 2$; $x^2 = (3.6, 5.4)$;

1). $\nabla F(x^2) = (16 - 4x_1, 12 - 2x_2) = (1.6, 1.2)$

2). h_2 = мах $f(3.6 + 1.6h, 5.4 + 1.2h) = 0.3$;

3). $x^3 = x^2 + h\,\nabla F(x^2) = (4.1, 5.8)$; $f(x^3) = 67.95$;

4). $\|x^3 - x^2\| > \varepsilon$, $k = k+1$.

Итерация 4. $k = 3$; $x^3 = (4.1, 5.8)$;

1). $\nabla F(x^2) = (16 - 4x_1, 12 - 2x_2) = (-0.4, 0.4)$

2). h_3 = мах $f(4.1 - 0.4h, 5.8 + 0.4h) = 0.31$;

3). $x^4 = x^3 + h\,\nabla F(x^3) = (3.98, 5.92)$; $f(x^4) = 67.99$;

4). $\|x^4 - x^3\| > \varepsilon$, $k = k+1$.

Итерация 5. $k = 4$; $x^4 = (3.98, 5.92)$;

1). $\nabla F(x^4) = (16 - 4x_1, 12 - 2x_2) = (0.04, 0.16)$

2). h_4 = мах $f(3.98 + 0.04h, 5.92 + 0.16h) = 0.29$;

3). $x^5 = x^4 + h\, VF(x^4) = (3.99, 5.97)$; $f(x^4) = 68.0$;

4). $\|x^5 - x^4\| < \varepsilon$ - оптимальное решение получено.

Итак, оптимальное решение $x^* = (3.99, 5.97)^т$; $f(x^*) = 68.0$.

7.3.2. Метод покоординатной оптимизации

(см.п.4.3.1)

Рассмотрим особенности применения алгоритма покоординатной оптимизации на примере рассмотренной ранее задачи (см. примеры 7.11, 7.12)

$f(x) = 16x_1 - 2x^2{}_1 + 12x_2 - x^2{}_2 \to \max$

Алгоритм заключается в последовательном выполнении итераций, каждая из которых состоит в выполнении ряда шагов. Задается начальная точка x^o; величина шага h; параметр дробления шага $\alpha\in(0,1)$, критерии окончания итерационного процесса некоторое малое $\varepsilon > 0$ и $\delta > 0$; номер цикла $c = 0$, номер итерации $k = 0$. Т.к. $x=(x_1, x_2)$, то $n = 2$.

<u>Пример 7.13.</u>

0). Исходное состояние. $x^o = (0,0)$; $h = 2$; $\alpha = 0.5$; $\varepsilon = 0.1$; $\delta = 0.1$; $c = 0$; $k = 0$; $n = 2$.

Итерация 1. $x^o = (0,0)$; $h = 1$; $c = 0$; $k = 0$;

1). Вычисляется номер проверяемой координаты вектора x на итерации k; $i(k) = k - c\,n + 1$; $i(0) = 1$;

2). Вычисляется значение функции

$f(x^k + h\,e_i(k)) = f(0 + 2, 0) = 24$; т.к. $e_1(0) = (1,0)$

т.к. $f(x^k + h\,e_i(k)) = 24 > f(x^k) = 0$, то $x^{k+1} = x^k + h\,e_i(k) = (2, 0)$,

3). $k = k + 1 = 1$; вычисляется $c' = [k/n] = 0$; т.к. $c' = c$, т.е. номер цикла не изменился, то производится переход на шаг 1.

Итерация 2. $x^1 = (2,0)$; $h = 2$; $c = 0$; $k = 1$;

1). $i(1) = k - c\,n + 1$; $i(1) = 2$; $e_2(1) = (0.1)$
2). $f(x^k+he_i(k)) = f(2,2) = 44 > f(x^k) = 24$, то $x^{k+1} = x^k+he_i(k) = (2,2)$,
3). $k = k + 1 = 2$; вычисляется $c' = 1$; т.к. $c' > c$, то $c = c' = 1$. Оценивается удачность цикла 0; цикл удачный, т.к. была удачная итерация; производится переход на шаг 4;
4). Т.к. $\|x^2 - x^1\| > \varepsilon$, то производится переход на шаг 1.

Итерация 3. $x^2 = (2, 2)$; $h = 2$; $c = 1$; $k = 2$;
1). $i(2) = k - c\,n + 1 = 1$;
2). $f(x^k + h\,e_i(k)) = f(4, 2) = 52 > f(x^k) = 44$, то $x^3 = (4, 2)$,
3). $k = k + 1 = 3$; вычисляется $c' = 1$;
т.к. $c' = c = 1$, производится переход на шаг 1;

Итерация 4. $x^3 = (4,2)$; $h = 2$; $c = 1$; $k = 3$;
1). $i(3) = k - c\,n + 1 = 2$;
2). $f(x^k + h\,e_i(k)) = f(4, 4) = 64 > f(x^k) = 52$, то $x^4 = (4, 4)$,
3). $k = k + 1 = 4$; вычисляется $c' = 2 > c$, то $c = c' = 2$.
Цикл удачный, т.к. была удачная итерация.
4). Т.к. $\|x^4 - x^3\| > \varepsilon$, то производится переход на шаг 1.

Итерация 5. $x^4 = (4, 4)$; $h = 2$; $c = 2$; $k = 4$;
1). $i(4) = k - c\,n + 1 = 1$;
2). $f(x^k + h\,e_i(k)) = f(6, 4) = 56 < f(x^k) = 64$, то
$f(x^k - h\,e_i(k)) = f(2, 4) = 56 < f(x^k) = 64$, то
$x^5 = x^4 = (4, 4)$ - итерация неудачна.
3). $k = k + 1 = 5$; вычисляется $c'= 2$;
т.к. $c' = c = 2$, производится переход на шаг 1;

Итерация 6. $x^5 = (4,4)$; $h = 2$; $c = 2$; $k = 5$;
1). $i(5) = k - c\,n + 1 = 2$;
2). $f(x^k + h\,e_i(k)) = f(4, 6) = 68 > f(x^k) = 64$, то $x^6 = (4, 6)$,
3). $k = k + 1 = 6$; вычисляется $c' = 3 > c$, то $c = c' = 3$.
Цикл удачный, т.к. была удачная итерация.
4). Т.к. $\|x^6 - x^5\| > \varepsilon$, то производится переход на шаг 1.

Итерация 7. $x^6 = (4, 6)$; $h = 2$; $c = 3$; $k = 6$;
1). $i(6) = k - c\,n + 1 = 1$;
2). $f(x^k + h\,e_i(k)) = f(6, 6) = 60 < f(x^k) = 68$, то
$f(x^k - h\,e_i(k)) = f(2, 6) = 60 < f(x^k) = 68$, то
$x^7 = x^6 = (4, 6)$ - итерация неудачна.
3). $k = k + 1 = 7$; вычисляется $c' = 3$; т.к. $c' = c = 3$, производится переход на шаг 1;

Итерация 8. $x^7 = (4,6)$; $h = 2$; $c = 3$; $k = 7$;
1). $i(7) = k - c\,n + 1 = 2$;
2). $f(x^k + h\,e_i(k)) = f(4, 8) = 64 < f(x^k) = 68$, то
$f(x^k - h\,e_i(k)) = f(4, 4) = 64 < f(x^k) = 68$, то

$x^8 = x^7 = (4, 6)$ - итерация неудачна.

3). $k = k + 1 = 8$; вычисляется $c' = 4 > c$, то $c = c' = 4$.
Цикл неудачный, т.к. не было удачных итераций.
Производится переход на шаг 5.

5). Производится дробление шага $h = \alpha\, h$, где $0 < \alpha < 1$.
Т.к. $h = 1 > \delta = 0.1$, то производится переход на шаг 1.

. . . .

Аналогично проводятся итерации 9-16. Все циклы с 4-го по 7-ой аналогичны 3-му циклу (итерации 7, 8) и все они неудачны. На каждом цикле происходит дробление шага пока не выполнится условие $h < \delta$. Оптимальное решение: $x^* = (4, 6)^т$; $f(x^*) = 68.0$.

7.3.3. Метод условного градиента

(см.п.4.4.1)

Рассмотрим особенности применения метода условного градиента на примере решения следующей задачи квадратичного программирования.

<u>Пример 7.14.</u>

Найти максимум нелинейной функции

$$f(x) = 16x_1 - 2x^2_1 + 12x_2 - x^2_2 \to \max$$

при ограничениях

$$\begin{aligned} 2x_1 - 3x_2 &\le 4 \\ 4x_1 + 3x_2 &\le 26 \\ -2x_1 + 1x_2 &\le 2 \\ x_1, x_2 &\ge 0. \end{aligned}$$

0). Исходное состояние; $x^o = (0, 0)$, $\varepsilon = 0.1$; $k = 0$.

Итерация 1. $x^o = (0, 0)$; $k = 0$.

1).Рассчитывается $\nabla^т F(x^o) = (16 - 4x_1, 12 - 2x_2) = (16, 12)$;

2).Решается задача линейного программирования вида

$x^o_* = \arg\max \nabla^т F(x^o)\, x$, $x \in \Delta$

$x^o_* = (2\ 6)^т$

x^o_* задает направление максимального приращения целевой функции из точки x^o.

3).Вычисляется величина шага в направлении, задаваемом x^o_*, для чего решается задача одномерной оптимизации

$$h_o = \arg\max_{0\le h\le 1} f(x^o+h(x^o_*-x^o))\ \arg\max_{0\le h\le 1} f(2h,6h) = 1.18, \text{ тогда } h_o=1.$$

4).Вычисляется значение $x^1 = x^o + h_o(x^o_* - x^o) = (2\ 6)^т$.

5).Т.к. $\|x^1-x^o\|>\varepsilon$, то $k=k+1=1$; производится переход на шаг 1.

Итерация 2. $x^1 = (2\ 6)$; $k = 1$.

1). $\nabla^{т}F(x^1) = (8, 0)$;

2). $x^1_* = \arg\max (8, 0)\ x,\ x\in\Delta$; $x^1_* = (5, 2)^{т}$

3). $h_1 = \arg\max f(2 + 3h, 6 - 4h),\ 0 \le h \le 1$; $h_1 = 0.35$

4). $x^2 = x^1 + h_1(x^1_* - x^1) = (3.06, 4.59)^{т}$.

5). Т.к. $\|x^2-x^1\| > \varepsilon$, то k=k+1=2; производится переход на шаг 1.

Итерация 3. $x^2 = (3.06, 4.59)$; k = 2.

1). $\nabla^{т}F(x^2) = (3.76, 2.82)$;

2). $x^2_* = \arg\max (5.36, 1.76)\ x,\ x\in\Delta$; $x^2_* = (2, 6)^{т}$

3). $h_2 = \arg\max f(3.06 + 1.06h, 4.59 - 1.41h),\ 0\le h\le 1$; $h_2 = 0$.

4). $x^3 = x^2 = (3.06, 4.59)^{т}$.

5). Т.к. $\|x^3 - x^2\| < \varepsilon$, то решение $x^2 = (3.06, 4.59)^{т}$ оптимально.

Итак, оптимальное решение $x^* = (3.06, 4.59)^{т}$; $f(x^*) = 64.24$.

Необходимость учета ограничений в задаче оптимизации привела к тому, что оптимальное решение и значение целевой функции достаточно сильно отличаются от тех, которые были получены при поиске безусловного экстремума (см. предыдущие примеры 7.11, 7.12, 7.13).

7.3.4. Метод возможных направлений

(см.п.4.4.2)

Метод возможных направлений предназначен для решения общей задачи выпуклого программирования. Алгоритм носит итерационный характер. Каждая итерация заключается в выполнении следующих шагов.

0). Задание начального приближения $x^o \in \Delta$, e > 0; k = 0.

1). Выбор возможного направления γ^k.

2). Вычисление максимально возможной величины шага $h^{м}_k$ в направлении γ^k.

3). Выбор шага h_k, $0 \le h \le h^{м}_k$.

4). Вычисление нового приближения x^{k+1}.

5). Если $\|x^{k+1}-x^k\| \le \varepsilon$ или $\|\nabla F(x^k)\| \le \varepsilon$ - решение оптимально.

<u>Пример 7.15.</u>

Рассмотрим особенности алгоритма возможных направлений на примере решения следующей задачи нелинейного программирования. Найти максимум функции

$$f(x) = 16x_1 - 2x^2_1 + 12x_2 - x^2_2 \to \max$$

при ограничениях

$$2x_1 - 3x_2 \le 4$$
$$2x_1 + 1x_2 \le 12$$
$$-3x_1 + 1x_2 \le 2$$

$x_1, x_2 \geq 0.$

0). Исходное состояние: $x^o = (0, 0)^т$; $\varepsilon = 0.1$; $k = 0$.

Итерация 1. $x^o = (0, 0)^т$; $k = 0$.

1). Выбор возможного направления осуществляется на основе решения задачи линейного программирования следующего вида

$$z \to \max,$$
$$-\nabla^т f(x^o)\, \gamma + z \leq 0,$$
$$\nabla^т \varphi_i(x^o)\, \gamma + z \leq 0, \ \ i \in I_o = \{r \mid \varphi_r(x^o) = b_r\},$$
$$0 \leq \gamma_j \leq 1, \ \ j = 1, \ldots, n.$$

т.к. $x^o = (0, 0)^т$, то как равенства выполняются ограничения $x_1 \leq 0$, $x_2 \leq 0$. Тогда задача выбора направления имеет вид

$$\begin{array}{lcl} z \to \max, & \text{или} & z \to \max, \\ -(16-4x_1, 12-2x_2)\, \gamma + z \leq 0 & & -16\gamma_1 - 12\gamma_2 + z \leq 0 \\ -\gamma_1 + z \leq 0 & & -\gamma_1 + z \leq 0 \\ -\gamma_2 + z \leq 0 & & -\gamma_2 + z \leq 0 \\ -1 \leq \gamma_j \leq 1, \ j = 1,2 & & -1 \leq \gamma_j \leq 1, \ j = 1,2. \end{array}$$

Решая данную задачу симплекс-методом, получаем решение $\gamma = (1, 1)$; $z = 1$.

2). Вычисляем $h^м$ в направлении γ.

$h^м = \arg\max \{ h \mid \varphi_i(h, h) = b_i, i = 1,2,3\}$,

$h^м = \max \{ -4, 4, -1 \} = 4$.

3). Выбор шага h_k, $0 \leq h_k \leq h^м$.

$h_k = \arg \max f(h, h) = 4.67 > h^м$, следовательно, $h_k = 4$.

4). $x^1 = x^o + h_k\, \gamma = (4, 4)$.

5). $\|x^1 - x^o\| > \varepsilon$ - переход на шаг 1.

Итерация 2. $x^1 = (4, 4)^т$; $k = 1$.

1).
$$z \to \max,$$
$$-4\gamma_2 + z \leq 0$$
$$2\gamma_1 + 1\gamma_2 + z \leq 0$$
$$-\gamma_1 + z \leq 0$$
$$-\gamma_2 + z \leq 0$$
$$-1 \leq \gamma_j \leq 1, \ \ j = 1,2.$$

Решение $\gamma = (-1, 0.4)$; $z = 1.6$.

2). $h^м = \arg\max \{ h \mid \varphi_i(4-h, 4+0.4h) = b_i, i = 1,2,3\}$

$h^м = \max \{ -2.5, 0, 2.94 \} = 2.94$.

3). $h_k = \arg\max f(4-h, 4+0.4h) = 0.43$, $h_k = 0.43$.

4). $x^2 = x^1 + h_k\, \gamma = (3.57, 4.17)$.

5). $\|x^2 - x^1\| > \varepsilon$ - переход на шаг 1.

Итерация 3. $x^2 = (3.57 \ \ 4.17)^т$; $k = 2$.

1). Так как x^2 - внутренняя точка, то направление совпадает с градиентом в этой точке.

$\gamma = (1.72,\ 3.66)$.

2). $h^м = \arg\max\{h \mid \varphi_i(3.57+1.72h,\ 4.17+3.66h)=b_i,\ i=1,2,3\}$,

$h^м = \max\{ -1.34,\ 0.10,\ -5.7 \} = 0.1$.

3). $h_k = \arg\max f(3.57+1.72h,\ 4.17+3.66h) = 0.42 > h^м$, следовательно, $h_k = 0.1$.

4). $x^3 = (3.73\ 4.54)$.

5). $\|x^3-x^2\| > \varepsilon$ - переход на шаг 1.

Итерация 4. $x^3 = (3.73\ \ 4.54)^т$; $k = 3$.

1).
$$
\begin{aligned}
& z \to \max,\\
& -1.08\gamma_1 - 2.92\gamma_2 + z \le 0\\
& 2\gamma_1 + 1\gamma_2 + z \le 0\\
& -\gamma_1 + z \le 0\\
& -\gamma_2 + z \le 0\\
& -1 \le \gamma_j \le 1,\ \ j = 1,2.
\end{aligned}
$$

Решение $\gamma = (-1,\ 0.79)$; $z = 1.21$.

2). $h^м = \arg\max\{ h \mid \varphi_i(3.73-h,\ 4.54+0.79h) = b_i,\ i = 1,2,3\}$,

$h^м = \max\{ -2.32,\ 0,\ 2.28 \} = 2.28$.

3). $h_k = \arg\max f(3.73-h,\ 4.54+0.79h) = 0.23$,

$h_k = 0.23$.

4). $x^4 = (3.5\ 4.72)$.

5). $\|x^{k+1}-x^k\| > \varepsilon$ - переход на шаг 1.

Итерация 5. $x^4 = (3.5\ 4.72)^т$; $k = 4$.

1). Так как x^4 - внутренняя точка, то направление совпадает с градиентом в этой точке.

$\gamma = (2\ 2.56)$.

2). $h^м = \arg\max\{h \mid \varphi_i(3.5+2h,\ 4.72+2.56h)=b_i,\ i=1,2,3\}$,

$h^м = \max\{ -3.03,\ 0.04,\ -2.26 \} = 0.04$.

3). $h_k = \arg\max f(3.5+2h,\ 4.72+2.56h) = 0.33 > h^м$, следовательно, $h_k = 0.04$.

4). $x^5 = (3.58\ 4.84)$.

5). $\|x^{k+1}-x^k\| > \varepsilon$ - переход на шаг 1.

. . . .

И т.д. повторяются итерации вида 4, 5. На 9 итерации получаем решение (3.35 5.30), при котором выполняется критерий окончания процесса 5. Тогда оптимальное решение $x^* = (3.35\ 5.30)^т$; $f(x^*) = 66.67$.

7.4. Дискретное программирование

Задачи дискретного (целочисленного) программирования характерны тем, что решение описывается вектором с целочисленными компонентами. Тогда задача имеет вид

$$x^* = \arg\max_{x \in \Delta} f(x),$$

где $\Delta \subseteq Z^n$ - n-мерное пространство векторов, компоненты которых являются целыми числами.

7.4.1. Метод отсечений Р.Гомори

(см.п.5.3)

Метод предназначен для решения линейных задач целочисленного и частично целочисленного программирования. Алгоритм основан на последовательном решении задачи линейного программирования без ограничений на целочисленность и вводе ограничений (отсечений), отсекающих нецелочисленную точку и оставляющих все целочисленные точки. Рассмотрим особенности алгоритма на примере решения следующей задачи.

<u>Пример 7.16.</u>

$$\begin{aligned}
&4x_1 + 6x_2 + 2x_3 \to \max\\
&2x_1 + 1x_2 + 1x_3 \le 6\\
&1x_1 + 2x_2 - 3x_3 \le 4\\
&x_1, x_2, x_3 \ge 0, \text{ целые.}
\end{aligned}$$

Приведем задачу к канонической форме

$$\begin{aligned}
&4x_1 + 6x_2 + 2x_3 \to \max\\
&2x_1 + 1x_2 + 1x_3 + x_4 = 6\\
&1x_1 + 2x_2 - 3x_3 + x_5 = 4\\
&x_1, \ldots, x_5 \ge 0, \text{ целые.}
\end{aligned}$$

Итерация 1.

1). Решаем задачу симплекс-методом. Получаем решение

2	29.6	4.40	0.80
3	1.6	0.40	-0.20
2	4.4	0.60	0.20

Решение нецелочисленно. Переход на шаг 2.

2). Строим отсечение Р.Гомори по 1-ой компоненте (см. ф. (5.12).

$$\sum_{j\in j_q} \{\alpha_{ij}\} x_{qj}. - x_{n+1} = \{\beta_i\} .$$

здесь $\alpha_{11} = 0.6$; $\alpha_{14} = 0.4$; $\alpha_{15} = -0.2$; $\beta_1 = 0.6$.

Тогда отсечение имеет вид

$0.6x_1 + 0.4x_4 + 0.8x_5 - x_6 = 0.6$ или $3x_1 + 2x_4 + 4x_5 - x_6 = 3$.

3). Формируем задачу линейного программирования

$$\begin{aligned}
&4x_1 + 6x_2 + 2x_3 \to \max\\
&2x_1 + 1x_2 + 1x_3 + x_4 = 6\\
&1x_1 + 2x_2 - 3x_3 + x_5 = 4\\
&3x_1 + 2x_4 + 4x_5 - x_6 = 3.\\
&x_1,...,x_6 \geq 0.
\end{aligned}$$

Итерация 2.

1). В результате решения получаем

2	29.0	4.40	0.80	-0.20
3	1.75	0.40	-0.20	0.05
2	4.25	0.60	0.20	-0.05
5	0.75	0.00	0.00	0.25

Значение целевой функции уменьшилось, но решение не целочисленно. Переход на шаг 2.

2). Строим отсечение Р.Гомори по 2-ой компоненте.

$\alpha_{21} = 0.25$; $\alpha_{24} = 0.5$; $\alpha_{26} = 0.05$; $\beta_2 = 0.25$.

Тогда отсечение имеет вид

$0.25x_1 + 0.5x_4 + 0.05x_6 - x_7 = 0.25$.

3). Формируем задачу линейного программирования

$$\begin{aligned}
&4x_1 + 6x_2 + 2x_3 \to \max\\
&2x_1 + 1x_2 + 1x_3 + x_4 = 6\\
&1x_1 + 2x_2 - 3x_3 + x_5 = 4\\
&3x_1 + 4x_4 + 2x_5 - x_6 = 3.\\
&0.25x_1 + 0.5x_4 + 0.05x_6 - x_7 = 0.25.\\
&x_1,...,x_7 \geq 0.
\end{aligned}$$

Итерация 3.

1). В результате решения получаем

3	28.0	4.40	0.80	-0.20	-4.00
3	2.0	0.40	-0.20	0.05	1.00
2	4.0	0.60	0.20	-0.05	-1.00
5	2.0	0.00	0.00	0.25	5.00
6	5.0	0.00	0.00	0.00	20.00

Решение целочисленно. Работа алгоритма заканчивается. Оптимальное решение исходной задачи $x^* = (0\ \ 4\ \ 2)^т$; $L^*=28$.

7.4.2. Метод "ветвей и границ"

(см.п.5.4)

Метод "ветвей и границ" объединяет группу методов, характеризующихся общей схемой решения задач дискретной оптимизации. На основе этого метода могут решаться как линейные, так и нелинейные задачи, особенность заключается в алгоритме вычисления оценок. Рассмотрим особенности применения метода на примере решения общей задачи линейного целочисленного программирования.

$4x_1 + 6x_2 + 2x_3 \rightarrow \max$

$2x_1 + 1x_2 + 1x_3 \le 6$

$1x_1 + 2x_2 - 3x_3 \le 4$

$x_1, x_2, x_3 \ge 0$, целые.

Для построения возможных ветвлений вначале целесообразно определить границы изменения переменных. Для данной задачи это $x_1 \le 3$, $x_2 \le 4$, $x_3 \le 6$.

<u>Пример 7.17.</u>

Решим данную задачу с использованием метода ветвей и границ.

0). Исходное состояние: список кандидатов $Q = \{\Delta\}$, где Δ-множество допустимых решений задачи; рекорд $L^* = -\infty$; решение $x^* = (-1, -1, -1)^T$.

Итерация 1. $Q = \{\Delta\}$; $L^* = -\infty$; $x^* = (-1, -1, -1)^T$; $q = 0$.

1). В списке Q всего один кандидат (Δ).

2). $g(\Delta) = 29.6$; x - нецелочисленно.

3). $\Delta = \Delta^1 \cup \Delta^2$; $\Delta^1 = \{ \Delta \mid x_1 < 2\}$

$\Delta^2 = \{ \Delta \mid x_1 \ge 2\}$

4). $Q \ne \varnothing$, производится переход на шаг 1.

Итерация 2. $Q = \{ \Delta^1, \Delta^2\}$; $L^* = -\infty$; $x^* = (-1,-1,-1)^T$; $q = 2$.

1). Выбираем Δ^1.

2). $g(\Delta^1) = 29.6$; x - нецелочисленно.

3). $\Delta^1 = \Delta^3 \cup \Delta^4$; $\Delta^3 = \{ \Delta \mid x_1 < 2, x_2 < 2\}$

$\Delta^4 = \{ \Delta \mid x_1 < 2, x_2 \ge 2 \}$

4). $Q \ne \varnothing$ - шаг 1.

Итерация 3. $Q = \{ \Delta^2, \Delta^3, \Delta^4 \}$; $L^* = -\infty$; $x^* = (-1,-1,-1)^T$; $q = 4$.

1). Выбираем Δ^4.

2). $g(\Delta^4) = 29.6$; x - нецелочисленно.

3). $\Delta^4 = \Delta^5 \cup \Delta^6$; $\Delta^5 = \{ \Delta \mid x_1 < 2, x_2 \ge 2, x_3 < 4\}$

$\Delta^6 = \{ \Delta \mid x_1 < 2, x_2 \ge 2, x_3 \ge 4 \}$

4). $Q \neq \varnothing$ - шаг 1.

Итерация 4. $Q = \{ \Delta^2, \Delta^3, \Delta^5, \Delta^6\}$; $L^* = -\infty$; $x^* = (-1,-1,-1)^т$; $q=6$.

1). Выбираем Δ^6.

2). $g(\Delta^6) = 20 = L^*$; $x^* = (0,2,4)^т$. Δ^6 исключается из Q.

4). $Q \neq \varnothing$ - шаг 1.

Итерация 5. $Q = \{ \Delta^2, \Delta^3, \Delta^5\}$; $L^* = 20$; $x^* = (0,2,4)^т$; $q=6$.

1). Выбираем Δ^2.

2). $g(\Delta^2) = 18.4 < L^*$; Δ^2 исключается из Q.

4). $Q \neq \varnothing$ - шаг 1.

Итерация 6. $Q = \{ \Delta^3, \Delta^5\}$; $L^* = 20$; $x^* = (0,2,4)^т$; $q=6$.

1). Выбираем Δ^3.

2). $g(\Delta^3) = 20 = L^*$; Δ^3 исключается из Q.

4). $Q \neq \varnothing$ - шаг 1.

Итерация 7. $Q = \{\Delta^5\}$; $L^* = 20$; $x^* = (0,2,4)^т$; $q=6$.

1). Выбираем Δ^5.

2). $g(\Delta^5) = 29.6$; x - нецелочисленно.

3). $\Delta^5 = \Delta^7 \cup \Delta^8$; $\Delta^7 = \{ \Delta | x_1 = 0, x_2 \geq 2, x_3 < 4 \}$

$\Delta^8 = \{ \Delta | x_1 = 1, x_2 \geq 2, x_3 < 4 \}$

4). $Q \neq \varnothing$ - шаг 1.

Итерация 8. $Q = \{\Delta^7, \Delta^8\}$; $L^* = 20$; $x^* = (0,2,4)^т$; $q=8$.

1). Выбираем Δ^7.

2). $g(\Delta^7) = 29.6$; x - нецелочисленно.

3). $\Delta^7 = \Delta^9 \cup \Delta^{10}$; $\Delta^9 = \{ \Delta \mid x_1 = 0, 2 \leq x_2 \leq 3, x_3 < 4 \}$

$\Delta^{10} = \{ \Delta \mid x_1 = 0, \quad x_2 = 4, x_3 < 4 \}$

4). $Q \neq \varnothing$ - шаг 1.

Итерация 9. $Q = \{\Delta^8, \Delta^9, \Delta^{10}\}$; $L^* = 20$; $x^* = (0,2,4)^т$; $q=10$.

1). Выбираем Δ^8.

2). $g(\Delta^8) = 24 = L^*$; $x^* = (1,3,1)^т$. Δ^8 исключается из Q.

4). $Q \neq \varnothing$ - шаг 1.

Итерация 10. $Q = \{\Delta^9, \Delta^{10}\}$; $L^* = 24$; $x^* = (1,3,1)^т$; $q=10$.

1). Выбираем Δ^9.

2). $g(\Delta^9) = 24 = L^*$; Δ^9 исключается из Q.

4). $Q \neq \varnothing$ - шаг 1.

Итерация 11. $Q = \{\Delta^{10}\}$; $L^* = 24$; $x^* = (1,3,1)^т$; $q=10$.

1). Выбираем Δ^{10}.

2). $g(\Delta^{10}) = 28 = L^*$; $x^* = (0,4,2)^т$; Δ^{10} исключается из Q.

4). $Q = \varnothing$ полученное решение $x^* = (0,4,2)^т$ оптимально.

Итак, оптимальное решение $x^* = (0,4,2)^т$; $L^* = 28$.

РАЗДЕЛ 2. ПРИНЯТИЕ РЕШЕНИЙ НА ОСНОВЕ ДИНАМИЧЕСКИХ МОДЕЛЕЙ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМ

8. ПОСТАНОВКА И КАЧЕСТВЕННЫЙ АНАЛИЗ ПРОБЛЕМЫ УПРАВЛЕНИЯ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

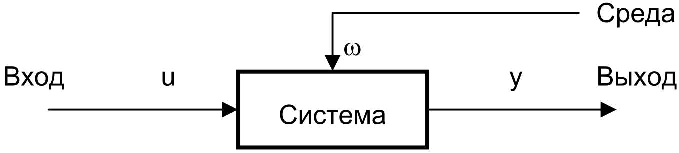
8.1.Содержание проблемы управления динамическими системами

Проектирование, внедрение и эксплуатация военно-технических систем связаны с необходимостью решения ряда проблем, центральными из которых являются проблемы повышения оперативности и качества принятия решений в указанных системах на различных этапах их жизненного цикла. Основным средством повышения качества решений в современных условиях является автоматизация данного процесса на базе разработки комплексов математических моделей, с помощью которых осуществляется исследование процессов функционирования основных элементов военно-технических систем. Выбор того или иного варианта математической модели - суть неформальная процедура, требующая разностороннего и глубокого системного анализа ситуации принятия решения. При этом исследователь должен выбрать исходную структуру (модель) принятия решения и в рамках этой структуры произвести формализованное описание решения, условий, ограничивающих выбор, и предпочтений, которым должно удовлетворять решение.

В качестве таких моделей в настоящее время принято рассматривать два существенно различных класса моделей: статические и динамические. И, если в статических моделях искомое решение не является функцией времени, т.е. время фиксировано, то в динамических моделях решение описывается некоторой функцией времени.

Функция принятия (выбора) решения представляет собой основу управления (u), которое направлено на достижение целей функционирования (выхода y) системы в различных условиях воздействия среды (ω).

Под управлением будем понимать процесс целенаправленного изменения выхода системы, осуществляемый путем воздействия на нее.



Далее будем полагать, что среда полностью известна (детерминированная), тогда процесс функционирования системы под воздействием управления можно описать отношением вида

$$\Gamma \subseteq T \times U \times Y,$$

где

T - множество моментов времени;

U - множество управляющих воздействий;

Y - множество выходов системы.

Недостатком данного отношения является то, что оно не отражает процессы, происходящие в самой системе. Вследствие этого для пары $(t,u) \in T \times U$ выход однозначно не определен, т.е. конкретному управляющему воздействию u в некоторый момент времени t могут соответствовать подмножества выходов из Y.

Введем понятие состояние системы, под которым будем понимать совокупность параметров (характеристик), отражающих наиболее существенные свойства системы и определяющих ее поведение (функционирование). Множество состояний системы обозначим через X, тогда функционирование системы можно описать отношением

$$G \subseteq T \times X \times U \times Y.$$

Изучение систем, описываемых данным отношением, основывается на выполнении принципа причинности:

1) выход системы не зависит от входных воздействий, которые могут иметь место в будущем (упорядоченность причинно-следственных связей);

2) выход системы однозначно определяется ее прошлыми состояниями и входными воздействиями (однозначность причинно-следственных связей);

Системы, удовлетворяющие принципу причинности, называются динамическими системами. Для таких систем отношение, описывающее ее, является отображением (терминальным) вида

$$g: T \times X \times U \to Y.$$

Для удобства исследования свойств системы терминальное отображение g, как правило, подвергается декомпозиции и представляется в виде

$$g = p \circ h,$$

где p: $T \times X \times U \to X$ - переходное отображение (функция);

h: $X \to Y$ - отображение (функция) выхода.

Таким образом исходная система разбивается на две подсистемы: подсистему, описывающую изменение состояния, такая подсистема называется ф у н д а м е н т а л ь н о й системой; и подсистему, описывающую выход системы. Исходная динамическая система, описываемая отображением g, в этом случае называется к а н о н и ч е с к о й системой.

Основное содержание процесса управления системой заключается в выработке управляющих воздействий u, которые обеспечивают достижение целей функционирования системы, описываемых выходом y(t). Основой выработки управлений u является наблюдение выхода y(t) и оценка x'(t) текущего состояния (далее будем полагать, что оценка совпадает с состоянием, т.е. x'(t) = x(t).

Процесс управления можно характеризовать схемой

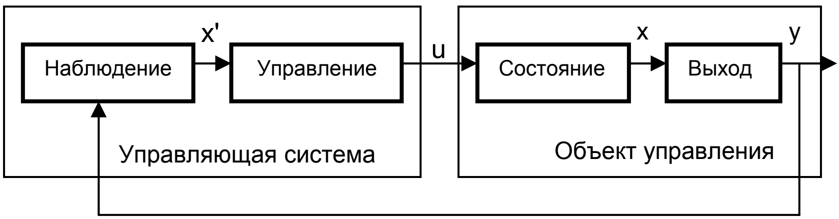


Рис. 8.1.

Совокупность управляющей системы и объекта управления часто называют с и с т е м о й у п р а в л е н и я.

Для систем управления характерно исследование трех групп проблем:

1) анализ качественных свойств системы таких как, например, устойчивость, наблюдаемость, управляемость,..;

2) оценивание состояния (или других неизвестных характеристик) динамической системы по результатам измерений - проблема идентификации;

3) проблема управления.

Проблемы принятия (выбора) решений связаны именно с третьей проблемой - проблемой управления, которая заключается в нахождении воздействия (управляющего) - (u), наилучшего в некотором смысле, и которое позволяет перевести динамическую систему из начального состояния в некоторое заданное конечное состояние, связанное с целевым назначением системы. Выбор (принятие) решений является основной функцией управления и основывается на формализованном описании ситуации - на модели принятия решения.

Одним из основных этапов разработки модели принятия решения, рассмотренных в (8.3), является выбор математической конструкции, описывающей решение и определяющей класс моделей, в рамках которых производится адекватное описание существа решаемых проблем. В динамических моделях систем решением является управление u, которое может выступать либо в виде функции времени $u = u(t)$ (такое управление, как правило, называют программным управленим), либо в виде функции состояния $u = u(t,x(t))$ (или функции выхода $u = u(t,y(t))$, которая называется синтезирующей функцией управления или управлением в форме синтеза (такое управление также часто называют управлением с обратной связью).

Далее во втором разделе будут рассмотрены методы и алгоритмы поиска программного управления, а в третьем разделе проблемы синтеза управления.

8.2. Основные понятия динамических систем

Современная теория управления динамическими системами получила интенсивное развитие во второй половине XX века (в значительной степени вследствие потребностей космической теории и практики). Именно в эти годы произошла определенная стантардизация терминологии и методик исследования систем с использованием динамических моделей. Вместе с тем теория управления базируется на результатах, полученных в теории регулирования, исследованиях дифференциальных уравнений, классическом вариационном исчислении, методах решения экстремальных задач. Для установления глубоких связей результатов, полученных в указанных областях знаний целесообразно определить основные понятия, которые будут использоваться в дальнейшем при рассмотрении алгоритмов решения задач.

8.2.1. Математические модели динамических систем

Основным инструментом принятия решений в военно-технических системах является модель, причем модель, построенная с использованием некоторых математических конструкций (математическая модель), создает основу для разработки формальной схемы(последовательности действий, алгоритма) поиска наилучшего решения. С точки зрения необходимого для принятия решения уровня описания природы процессов, протекающих в системе, можно различать, по крайней мере, два существенно различных класса моделей.

Динамические модели - описывают причинно-следственные связи изменений состояния системы и процессов, влияющих на это изменение.

Статические модели - описывают связь между состоянием системы и другими характеристиками в некоторый фиксированный момент (моменты) времени.

В статических моделях решение описывается некоторым вектором (элементом линейного или векторного пространства), и для поиска наилучшего решения используются методы выбора альтернатив в конечномерном пространстве векторов - методы математического программирования.

В динамических моделях решение описывается функцией (векторной функцией) времени, и выбор наилучших решений основывается на методах, развиваемых в рамках теории оптимального управления. Существенным в динамических моделях, по сравнению со статическими моделями, является также, то, что здесь необходимо вводить понятие состояние системы - совокупность параметров, отражающих наиболее существенные свойства системы и определяющих ее поведение.

В теории оптимального управления под понятием "динамическая система" традиционно понимается динамическая модель системы. Здесь следует отметить, что понятие "система" чрезвычайно многообразно. Системой называют реально существующие физические системы, системы дифференциальных уравнений, социальные системы и другие. В дальнейшем, следуя традиционной терминологии и говоря о динамической системе, мы будем иметь в виду именно динамическую модель системы, что впрочем, как правило, ясно из контекста.

Итак, динамическая система описывается отображениями (функциями)

$$p: T \times X \times U \to X, \qquad (8.1)$$
$$h: X \to Y . \qquad (8.2)$$

Рассмотрим некоторые определения динамических систем, связанные со свойствами, используемых в (8.1), (8.2) образующих множеств:

- г о м о г е н н а я динамическая система - система, в которой множества U, X, Y имеют одинаковую мощность;
- динамическая система называется системой с н е п р е р ы в н ы м временем, если T - подмножество вещественных чисел ($T \subseteq R$);
- динамическая система называется системой с д и с к р е т н ы м временем, если T - подмножество натуральных чисел ($T \subseteq N$);
- динамическая система называется:
  - к о н е ч н о й системой, если множество состояний X- конечно;
  - с ч е т н о й системой, если множество состояний X- счетно;
  - к о н е ч н о м е р н о й системой, если множество X- конечномерно, т.е. $X \subseteq R^n$, тогда n - порядок системы;
  - б е с к о н е ч н о м е р н о й системой, если множество X - бесконечномерное линейное пространство;
- динамическая система называется с т а ц и о н а р н о й, если ее состояние не зависит от того, в какой момент времени осуществлялось входное воздействие, а зависит только от вида воздействия (состояние инвариантно относительно сдвига во времени входного воздействия);
- динамическая система называется с в о б о д н о й системой, если $U = \{0\}$.

Динамическая система называется д и ф ф е р е н ц и а л ь н о й системой, если переходная функция является решением задачи Коши $x(t) = p(t,t_o,x_o,u)$ для системы дифференциальных уравнений, $x_o=x(t_o)$ - состояние динамической системы в начальный момент времени t_o.

$$\dot{x}(t) = \varphi (t, x, u), \qquad (8.3)$$

здесь $x(t) \in R^n$, $u(t) \in R^m$, $t \in (t_o, t_f]$.

Система дифференциальных уравнений (8.3), описывающих изменение состояния, называется н о р м а л ь н о й системой дифференциальных уравнений.

Далее под термином "динамическая система" будем понимать конечномерную дифференциальную динамическую систему с непрерывным временем.

Довольно часто в задачах оптимального управления различают класс автономных динамических систем, т.е. систем, описываемых уравнениями, в которых функции правых частей не зависят явно от времени, тогда (8.3) можно переписать

$$\dot{x}(t) = \varphi(x, u).$$

Вообще говоря, расширив вектор состояния путем ввода дополнительной переменной $x_o = t$, можно неавтономную систему свести к некоторой автономной.

Динамическая система (8.3) называется линейной, если все функции $\varphi_i(t,x,u)$, $i=1,...,n$ линейны относительно x и u, тогда (8.3) можно переписать в виде

$$\dot{x}(t) = A(t)x + B(t)u + C(t), \tag{8.4}$$

где $A(t)$ и $B(t)$ - матричные функции.

Линейная динамическая система является стационарной, если матричные функции $A(t)$, $B(t)$ и $C(t)$ не зависят от времени, т.е. $A(t)=A$; $B(t)=B$; $C(t)=C$. Тогда (8.4) имеет вид

$$\dot{x}(t) = A x + B u + C, \tag{8.5}$$

Итак, изменение состояния $x(t)$ динамической системы на интервале времени $t\in(t_o,t_f]$ из некоторого начального состояния $x(t_o)$ под воздействием управления $u(t)$ описывается системой дифференциальных уравнений

$$\dot{x}(t) = \varphi(t, x, u),$$

$$x(t) \in X \subseteq R^n,\ u(t) \in U \subseteq R^m,\ t \in T = (t_o, t_f].$$

Здесь

X - множество состояний (фазовое пространство) динамической системы; $X \subseteq R^n$;

$x(t) = (x_1(t),..,x_n(t))^T$ - вектор состояния (фазовый вектор) в момент времени t; $x_i(t)$ - i-я компонента вектора состояния, $i = 1,...,n$;

$x(t)$, $t\in(t_o,t_f]$ - фазовая траектория динамической системы на интервале времени $(t_o,t_f]$.

U - множество допустимых управлений динамической системы; $U \subseteq R^m$;

$u(t) = (u_1(t),...,u_m(t))^T$ - вектор значений управления (мгновенных) в момент времени t;

$u(t)$, $t \in (t_o,t_f]$ - у п р а в л е н и е (программа управления) динамической системой на интервале времени $(t_o,t_f]$.

$T = (t_o,t_f]$ - и н т е р в а л у п р а в л е н и я (множество моментов времени) динамической системой.

Дифференциальные уравнения (8.3), описывающие скорость изменения компонент вектора состояния, часто называют у р а в н е н и я м и с о с т о я н и я.

Дифференциальные уравнения, в которых состояние x является функцией одного аргумента $x=x(t)$, называются о б ы к н о в е н н ы м и дифференциальными уравнениями. В противном случае (если x-функция нескольких аргументов) они называются дифференциальными уравнениями в ч а с т н ы х п р о и з в о д н ы х.

П о р я д о к дифференциального уравнения равен порядку наивысшей производной, встречающейся в уравнении. Если рассматривается система дифференциальных уравнений первого порядка, то п о р я д к о м с и с т е м ы считается количество уравнений в системе (размерность вектора состояния). Следует отметить, что решение дифференциального уравнения n-го порядка можно свести к решению системы (n) дифференциальных уравнений первого порядка.

Далее будем полагать, что динамическая система описывается системой из (n) обыкновенных дифференциальных уравнений первого порядка.

8.2.2. Решение дифференциальных уравнений состояния.

Решение уравнений, описывающих изменение состояния (дифференциальных уравнений)

$$\dot{x}(t) = \varphi(t, x, u),$$

при заданных начальных условиях $x_o=x(t_o)$ и известном управлении $u(t)$, $t \in (t_o,t_f]$ представляет собой функцию

$$x(t) = p(t,t_o,x_o,u),$$

которая описывает состояние в любой момент времени t. Задача поиска такой функции, являющейся решением системы дифференциальных уравнений, известна как з а д а ч а К о ш и.

Так как дифференциальные уравнения представляют собой модель динамической системы, то первым вопросом, который возникает при анализе данной модели, является следующий "Существует ли допустимое состояние системы в момент времени t, в которое она переходит под воздействием управления u(t), и единственно ли это состояние ?". Ответ на этот вопрос дается теоремой о существовании и единственности решения задачи Коши, которая описывает каким ограничениям должны удовлетворять функции $\varphi(t,x,u)$, $u(t)$ для того, чтобы существовало единственное решение.

Теорема (существования и единственности).
Пусть задана система дифференциальных уравнений

$$\dot{x}(t) = \varphi (t, x, u) , \qquad (8.6)$$

где $u(t)$, $t \in T$ является кусочно-непрерывной функцией; функция $\varphi(t,x,u)$ определена в области $A \subseteq T \times X \times U$, непрерывна в ней по t, а по x удовлетворяет условию

$$\| \varphi (t, x_1, u) - \varphi (t, x_2, u) \| \leq L \| x_1 - x_2 \| , \qquad (8.7)$$

для $\forall x_1, x_2 \in X$; $u \in U$; $t \in T$, где L - постоянная, больше 0. Тогда можно указать интервал времени $\theta \subseteq T$, $t_o \in \theta$, на котором существует и при том единственное решение $x = \xi(t)$, системы уравнений (8.6), удовлетворяющее начальным условиям $x = \xi(t_o) = x_o$.

Условие (8.7) называют условием Л и п ш и ц а; L - постоянная Липшица. Это условие являются более строгим, чем условие непрерывности функции, но менее строгим, чем условие непрерывности ее частных производных.

В ряде практических приложений функция $\varphi(t,x,u)$ является разрывной по t в счетном множестве точек, тогда рассмотренная теорема применима на участках непрерывности функции правых частей (8.6). Решение $\xi(t)$ в этом случае является непрерывной функцией, которая в точках разрыва может иметь изломы.

В общем случае уравнения состояния являются нелинейными и решение задачи Коши производится на основе численных методов. Наиболее известными и часто используемыми здесь являются методы численного решения системы дифференциальных уравнений: одношаговые методы (например, методы Рунге-Кутта); конечно-разностные методы (например, методы Адамса);

или комбинированные методы, объединяющие достоинства конечно-разностных и одношаговых.

Рассмотрим решение уравнений состояния линейной динамической системы

$$\dot{x}(t) = A(t)x + B(t)u . \qquad (8.8)$$

Наряду с линейной системой дифференциальных уравнений вида (8.8) будем рассматривать о д н о р о д н у ю линейную систему дифференциальных уравнений

$$\dot{x}(t) = A(t)\,x . \qquad (8.9)$$

Утверждение.

Совокупность всех решений однородной системы из n линейных дифференциальных уравнений образует линейное (векторное) пространство размерности n.

Тогда в этом n-мерном пространстве решений (а каждое решение описывается вектором) существует максимальная система линейно независимых решений (базис) $S_o = \{\xi_1(t), \ldots, \xi_n(t)\}$. Такая система S_o называется ф у н д а м е н т а л ь н о й системой решений и любое решение $\xi(t)$ однородной системы дифференциальных уравнений (8.9) может быть выражено как линейная комбинация решений из, S_o, т.е.

$$\xi(t) = \alpha_1\xi_1(t), \ldots, \alpha_n\xi_n(t), \qquad (8.10)$$

где $\alpha_1,\ldots,\alpha_n \in R$. Выражение (8.10) называется о б щ и м решением системы (8.9). Иначе (8.10) можно переписать

$$\xi(t) = S(t)\,\alpha, \qquad (8.11)$$

где $\alpha = (\alpha_1, \ldots, \alpha_n)^T$ - вектор коэффициентов, а $S(t)$ - матрица размерности $n\times n$, столбцы которой образуют векторы решений $\xi_i(t)$, $i=1,\ldots,n$

$$S(t) = \begin{Vmatrix} \xi_{11}(t) & \xi_{21}(t) & ,\ldots, & \xi_{n1}(t) \\ \cdot & \cdot & \cdot & \cdot \\ \xi_{1n}(t) & \xi_{2n}(t) & ,\ldots, & \xi_{nn}(t) \end{Vmatrix}$$

Такая матрица решений может быть составлена для любого момента времени $t\in T$, в том числе и для t_o. Так как матрица

$S(t_o)$ состоит из линейно независимых векторов, то существует обратная к ней матрица $S^{-1}(t_o)$.

Определим матрицу $\Phi(t,t_o)$ как

$$\Phi(t,t_o) = S(t)\, S^{-1}(t_o). \tag{8.12}$$

Эта матрица также образована из векторов фундаментальной системы решений, но отличие ее в том, что в момент времени t_o она образует в соответствии с (8.12) единичную матрицу $\Phi(t_o,t_o) =$ E.

Матрица $\Phi(t,t_o)$ называется ф у н д а м е н т а л ь н о й или п е р е х о д н о й матрицей однородной системы дифференциальных уравнений и имеет большое значение при исследовании динамических систем и процессов управления ими. Рассмотрим свойства фундаментальной матрицы.

$$1.\ \Phi(t_o,t_o) = E. \tag{8.13}$$

$$2.\ \det \Phi(t,t_o) \neq 0,\ \forall t \in T. \tag{8.14}$$

Свойства 1, 2 непосредственно следуют из определения фундаментальной матрицы (8.12).

3. $\Phi(t,t_o)$ - решение системы дифференциальных уравнений (8.9), т.е.

$$\frac{d\Phi(t,t_o)}{dt} = A(t)\ \Phi(t,t_o)\,. \tag{8.15}$$

Действительно,

$$\frac{d\Phi(t,t_o)}{dt} = \frac{dS(t)}{dt} S^{-1}(t_o) = \left\| \frac{d\xi_1(t)}{dt} \ldots \frac{d\xi_n(t)}{dt} \right\| S^{-1}(t_o) =$$

$$\left\| A\,\xi_1(t), \ldots, A\,\xi_n(t) \right\| S^{-1}(t_o) = A\,S(t)\,S^{-1}(t_o) = A\ \Phi(t,t_o)$$

4. Решение $x(t)$, удовлетворяющее начальным условиям $x(t_o)$, можно записать в виде

$$x(t) = \Phi(t,t_o)\, x(t_o)\,. \tag{8.16}$$

В самом деле выполнение начальных условий следует из равенства

$$x(t_o) = \Phi(t_o,t_o)\, x(t_o).$$

Кроме того, $x(t) = \Phi(t,t_o)\, x(t_o)$ является решением уравнения (8.9)

$$\frac{dx}{dt} = \frac{d\Phi(t,t_o)}{dt}\, x(t_o) = A(t)\, \Phi(t,t_o)\, x(t_o) = A(t)\, x(t).$$

5. $\Phi(t,t_o) = \Phi(t,\tau)\,\Phi(\tau,t_o)$.

6. $\Phi(t,t_o) = \Phi^{-1}(t_o,t)$.

Действительно, $\Phi(t,t_o) = S(t)\,S^{-1}(t_o) = (S(t_o)\,S^{-1}(t))^{-1} = \Phi^{-1}(t_o,t)$.

Решение неоднородной системы дифференциальных уравнений (8.8) определяется в соответствии с ф о р м у л о й К о ш и

$$x(t) = \Phi(t,t_o)\,x(t_o) + \Phi(t,t_o)\int_{t_0}^{t}\Phi^{-1}(\tau,t_o)\,B(\tau)\,u(\tau)\,d\tau\,. \qquad (8.16)$$

Поясним последнее выражение, для этого произведем замену неизвестных функций

$$x(t) = \Phi(t,t_o)\,y. \qquad (8.17)$$

Тогда

$$\frac{dx}{dt} = \frac{d\Phi(t,t_o)}{dt}\,y + \Phi(t,t_o)\,\frac{dy}{dt} =$$

$$= A(t)\,\Phi(t,t_o)\,y + \Phi(t,t_o)\,\frac{dy}{dt} = A(t)\,\Phi(t,t_o)\,y + B(t)\,u(t)\,,$$

откуда

$$\frac{dy}{dt} = \Phi^{-1}(t,t_o)\,B(t)\,u(t)\,. \qquad (8.18)$$

Из (8.17) следует, что $y(t) = \Phi^{-1}(t_o,t_o)\,x(t_o)$. Тогда, интегрируя дифференциальное уравнение (8.18) при начальных условиях $y(t_o)$, получим

$$y(t) = x(t_o) + \int_{t_0}^{t}\Phi^{-1}(\tau,t_o)\,B(\tau)\,u(\tau)\,d\tau\,.$$

Подставляя последнее выражение в (8.17), получим формулу Коши (8.16).

Для нахождения переходной матрицы $\Phi(t,t_o)$ можно (n) раз численно решать задачу Коши для системы дифференциальных уравнений (8.9) при начальных условиях $x(t_o) = e_i$, $i = 1,....,n$, где e_i - n-мерный вектор, все компоненты которого, кроме i-ой, равны 0 (i-ая компонента равна 1). Полученные решения $x_1(t),..., x_n(t)$ являются столбцами переходной (фундаментальной) матрицы $\Phi(t,t_o)$.

Переходная матрица свободной стационарной динамической системы, описываемой однородными дифференциальными уравнениями с постоянными коэффициентами (8.9),

$$\dot{x}(t) = A\,x$$

представляет собой экспоненту от At, т.е. (рассмотрим случай, когда $t_o = 0$)

$$\Phi(t,\ t_o) = \Phi(t - t_o) = \Phi(t) = e^{At}, \tag{8.19}$$

где $$e^{At} = \sum_{k=0}^{\infty} A^k\,(t^k/\ t!)\ . \tag{8.20}$$

Итак, если уравнения состояния являются нелинейными, то состояние x(t), в которое перейдет система из некоторого начального состояния $x(t_o)$, рассчитывается на основе численных методов. Для линейной системы рассчитывается переходная матрица $\Phi(t,t_o)$, и состояние в момент t определяется как $x(t) = \Phi(t,t_o)\,x(t_o)$ для любого начального состояния $x(t_o)$. В этой связи довольно часто для удобства исследования систем производится ее линеаризация, с заменой нелинейных уравнений состояния линейными уравнениями. В настоящее время при исследовании динамических систем наиболее широко используются два способа линеаризации.

1-ый способ - пошаговая линеаризация. Весь интервал управления $(t_o,t_f]$ разбивается на интервалы $(t_i,\ t_{i+1}]$, $i = 0,..,\ N-1$ меньшей длительности, и на каждом таком интервале система описывается линейными дифференциальными уравнениями,для которых находится переходная матрица $\Phi_i(t,t_i)$. Тогда состояние в момент t можно определить как

$$x(t) = \Phi_i(t,t_i)\,x(t_i), \quad t \in (t_i,\ t_{i+1}].$$

2-ой способ - линеаризация вблизи опорной траектории. В этом случае $x(t) = x^o(t) + \delta x(t)$, где $x^o(t)$ - опорная траектория, $\delta x(t)$ - отклонение от опорной траектории. Аналогично, $u(t) = u^o(t) + \delta u(t)$. Тогда нелинейную систему дифференциальных уравнений (8.3) можно переписать

$$\frac{d}{dt}(x^o(t) + \delta x(t)) = \frac{d\,x^o(t)}{dt} + \frac{d\,\delta x(t)}{dt} =$$

$$= \varphi\,(\,t,x^o,u^o) + F_x(\,t,x^o,u^o)\,\delta x(t) + F_u(\,t,x^o,u^o)\,\delta u(t) + \delta F(t)\ .$$

Здесь $F_x(t,x^o,u^o) = \|\partial\varphi_i/\partial x_j\|$ - матрица Якоби по x; $F_u(t,x^o,u^o) = \|\partial\varphi_i/\partial u_j\|$ - матрица Якоби по u; $\delta F(t)$ - остаточный член. Пренебрегая в последнем выражении остаточным членом $\delta F(t)$, можно записать дифференциальные уравнения для отклонений от опорной траектории

$$\frac{d\,\delta x(t)}{dt} = A^o(t)\,\delta x(t) + B^o(t)\,\delta u(t))\ .$$

Матрицы $A^o(t)$ и $B^o(t)$ зависят от опорной траектории, соответственно, и переходная матрица $\Phi^o(t,t_o)$ будет справедлива при прогнозе движения линеаризованной динамической системы вблизи конкретной опорной траектории.

8.2.3. Сопряженная система

Наряду с системой дифференциальных уравнений (8.3)

$$\frac{d\,x_i(t)}{dt} = \varphi_i(\,t,\,x,\,u),\ i = 1,\dots,n\ ,$$

описывающих изменение состояния динамической системы, в задачах оптимального управления часто рассматривают с о п р я ж е н н у ю систему дифференциальных уравнений

$$\frac{d\psi_i(t)}{dt} = -\sum_{j=1}^{n}\frac{\partial\varphi_j(t,x,u)}{\partial x_i}\psi_j(t),\ i = 1,\dots,n, \tag{8.21}$$

Если исходная система дифференциальных уравнений линейная, т.е. имеет вид (8.8), то в соответствии с (8.21) сопряженная система имеет вид

$$\dot\psi\,(t) = -\,A^T\,\psi. \tag{8.22}$$

Система (8.22) представляет собой однородную систему дифференциальных уравнений, переходную матрицу этой системы обозначим как $\Psi(t,t_o)$. Аналогично (8.15), матрица $\Psi(t,t_o)$ является решением системы (8.22). Тогда

$$\frac{d}{dt}\left(\Psi^T(t,t_o)\Phi(t,t_o)\right) = \frac{d\Psi^T(t,t_o)}{dt}\Phi(t,t_o) + \Psi^T(t,t_o)\frac{d\Phi(t,t_o)}{dt} =$$

$$= (\,-\,A^T(t)\,\Psi(t,t_o))^T\,\Phi(t,t_o) + \Psi^T(t,t_o)\,A(t)\,\Phi(t,t_o)\ =$$

$$=\ -\,\Psi^T(t,t_o)\,A(t)\,\Phi(t,t_o) + \Psi^T(t,t_o)\,A(t)\,\Phi(t,t_o)\ = 0\ .$$

Таким образом, $\Psi^T(t,t_o)\ \Phi(t,t_o) = \mathrm{const},\ \forall t\in(t_o,t_f]$. Но, поскольку в момент $t=t_o$, $\Psi^T(t_o,t_o)\ \Phi(t_o,t_o) = E$, то $\Psi^T(t,t_o)\ \Phi(t,t_o) = E,\ \forall t\in(t_o,t_f]$. Тогда

$$\Psi^T(t,t_o) = \Phi^{-1}(t,t_o) = \Phi(t_o,t),\ \ \forall t\in(t_o,t_f]. \tag{8.23}$$

Рассмотрим, как изменяется скалярное произведение векторов состояний прямой и сопряженной систем

$$\frac{d}{dt}\left(\psi^{т}(t)x(t)\right) = \frac{d\psi^{т}(t)}{dt}x(t) + \psi^{т}(t)\frac{dx(t)}{dt} =$$

$$= (- A^{т}(t)\ \psi\ (t))^{т}\ x(t) + \psi^{т}(t)\ (A(t)\ x(t) + B(t)\ u(t)) =$$

$$= - \psi^{т}(t)\ A(t)\ x(t) + \psi^{т}(t)\ A(t)\ x(t) + \psi^{т}(t)\ B(t)\ u(t) =$$

$$= \psi^{т}(t)\ B(t)\ u(t).$$

Таким образом, в условиях отсутствия управления скалярное произведение векторов состояний прямой и сопряженной систем постоянно.

8.3. Постановка задач оптимального управления

Принятие решения в военно-технических системах, когда решение формализуется (математически описывается) векторной функцией времени, основывается на методах, развитых в современной теории оптимального управления. Данная теория начала интенсивно развиваться в конце сороковых, начале пятидесятых годов XX столетия. Следует отметить, что побудительным толчком активного проведения данных исследований явились, в первую очередь, потребности развивающейся ракетно-космической теории и практики. После опубликования результатов работ, выполненных школой Л.С.Понтрягина [75], произошла определенная стандартизация терминологии и методики решения задач в рамках новой научной дисциплины, которая получила название теории оптимального управления. В настоящее время данная идеология глубоко проникла во все сферы исследований и конструкторских разработок военно-технических систем. В этой связи в данном пособии при изложении вопросов принятия решений на динамических моделях будем придерживаться существующих понятий и терминов.

Вместе с тем с целью рассмотрения на единой концептуальной основе задач принятия решений, использующих модели различной природы, рассмотрим обобщенную модель выбора (см. 1.4), которая может быть представлена как

$$(Q(s), \Delta, \{ r_i, i \in C\}, f) . \qquad (8.24)$$

Здесь

Q(s) - исходная структура выбора (модель) типа s. Данная структура позволяет ставить задачи выбора, связанные с теми или иными структурными (модельными) ограничениями, задаваемыми посредством, например, дифференциальных уравнений для динамических систем или посредством сетей, алгебраических уравнений для статических систем.

Δ - пространство альтернатив (решений). В зависимости от типа исходной структуры выбора Q(s) это некоторое конечномерное пространство векторов или пространство векторных функций, отдельный элемент которого характеризует структуру решения (альтернативы), состав и смысл компонент этой альтернативы.

$\{r_i, i \in C\}$ - множество отношений, ограничивающих выбор; C - множество индексов отношений, ограничивающих выбор. Данные отношения задаются на Δ и вводятся непосредственно при постановке задач выбора, они отражают основные ограничения, связанные с процессом функционирования рассматриваемых систем.

f - отношение предпочтения, задаваемое на множестве альтернатив Δ и отражающее требования, предъявляемые к оптимальному решению.

Для рассматриваемых здесь задач оптимального управления исходная структура выбора Q(s) задается системой дифференциальных уравнений

$$\dot{x}(t) = \varphi(t, x, u),$$

Пространство альтернатив Δ представляет собой декартово произведение базовых множеств $\Delta = T \times X \times U$, где T - множество моментов времени; X - множество состояний системы; U - множество управлений.

Множество отношений, ограничивающих выбор, $\{r_i, i \in C\}$ задает на Δ множество допустимых альтернатив $G = G_t \times G_x \times G_u \subseteq \Delta$.

Отношение предпочтения f, как правило, задается с использованием функций (целевых функций, функционалов) вида $f: \Delta \to R^1$ или $f: T \times X \times U \to R^1$.

Таким образом, элементами Δ является тройки вида (t,x,u). На самом деле выбирается пара функций (x(t),u(t)), причем в соответствии со структурой выбора, которая задается системой дифференциальных уравнений и, в силу теоремы существования и единственности, программа управления $u(t)$ при заданном $x(t_o)$ однозначно определяет фазовую траекторию $x(t)$, $\forall t \in T$, т.е. искомое решение однозначно определяется функцией $u(t)$. Следовательно, ограничения в задачах оптимального управления могут задаваться на $T \times X \times U$, а решением данной задачи является программа управления $u(t)$. В этих условиях отображение $f: T \times X \times U \to R^1$ представляет собой функционал $J(t,x,u)$ или $J(x(t),u(t))$, действующий из пространства функций (состояния и управления) в множество действительных чисел, и, в соответствии со сложившейся в теории оптимального управления практикой, ищется минимум этого функционала.

Тогда, в достаточно общем виде, задача оптимального управления может быть сформулирована следующим образом.

Необходимо найти векторную функцию управления $u(t)$, $t \in T$, доставляющую минимум функционалу

$$J(x,u) \to \min \tag{8.25}$$

при управлении динамической системой

$$\dot{x}(t) = \varphi(t, x, u), \tag{8.26}$$

которое удовлетворяет ограничениям вдоль траектории

$$(t,x,u) \in G = G_t \times G_x \times G_u \subseteq \Delta = R^1 \times R^n \times R^m, \tag{8.27}$$

при переводе системы из начального состояния $x(t_o)$ в конечное состояние $x(t_f)$

$$x(t_o) \in E_o \subseteq R^n,\ x(t_f) \in E_f \subseteq R^n . \tag{8.28}$$

Итак, в постановке задачи оптимального управления присутствует четыре элемента.

1). Функционал: $J(x, u) \to \min$.

2). Дифференциальные связи (динамическая система): $\dot{x}(t) = \varphi(t, x, u)$.

3). Ограничения вдоль траектории: $(t,x,u) \in G = G_t \times G_x \times G_u$.

4). Краевые условия: $x(t_o) \in E_o$, $x(t_f) \in E_f$.

Конкретизация указанных четырех элементов порождает различные типы задач. Рассмотрим варианты задания указанных элементов постановки задачи оптимального управления.

8.3.1. Способы задания функционала.

В зависимости от способа задания различают несколько видов функционалов, а, соответственно, задач оптимального управления:

З а д а ч а   Л а г р а н ж а (интегральный функционал), качество управления характеризуется некоторыми интегральными на всем интервале управления характеристиками, тогда функционал имеет вид

$$J(x,u) = \int_{t_0}^{t_f} F(t,x,u)\,dt \rightarrow \min \tag{8.29}$$

где F - некоторая дифференцируемая функция своих аргументов. Простейшим видом задачи Лагранжа является задача, в которой $F(t,x,u) = 1$. Тогда

$$J(x,u) = t_f - to \rightarrow \min,$$

т.е. задача становится задачей   м а к с и м а л ь н о г о   б ы с т р о д е й с т в и я.

З а д а ч а   М а й е р а (терминальный функционал), качество управления определяется значением фазовых переменных, которые достигаются на концах траектории.

$$J(x,u) = \Phi(\,t_o,\ x(t_o),\ t_f,\ x(t_f)) \rightarrow \min\ . \tag{8.30}$$

Например, функционал данного вида может характеризовать степень приближения конечного состояния динамической системы к заданному состоянию $x^{зад}(t_f)$. Тогда

$$J(x,u) = \Phi(\,t_f,\ x(t_f)) = (x^{зад}\,(t_f) - x(t_f))^2 \rightarrow \min.$$

Формально задачу Лагранжа можно представить в виде задачи Майера, действительно, введем новую фазовую переменную

$$x_o = J(x,u),$$

тогда

$$\dot{x}_o(t) = F(\,t,\ x,\ u\,)\ ,$$

и расширенный фазовый вектор

$$\overline{x}(t) = (\,x_o\ x_1\ ,\ldots,\ x_n\,)^т\ .$$

Задача Майера для расширенного фазового вектора $\overline{x}(t)$ и с функционалом

$$J(\overline{x}, u) = \Phi(t_f, \overline{x}(t_f)) = x_o(t_f) \rightarrow \min$$

является аналогом задачи Лагранжа (8.29).

З а д а ч а Б о л ь ц а (смешанный функционал), качество управления определяется функционалом смешанного типа, который имеет интегральную и терминальную составляющие

$$J(x,u) = \Phi(t_o, x(t_o), t_f, x(t_f)) + \int_{t_0}^{t_f} F(t,x,u)\,dt \rightarrow \min . \qquad (8.31)$$

Отметим, что данную задачу также можно представить в форме задачи Майера.

8.3.2. Виды дифференциальных связей.

Вид дифференциальных уравнений, описывающих динамическую систему, представляет собой существенную характеристику задачи оптимального управления. В 8.2 были рассмотрены различные динамические системы и особенности дифференциальных уравнений, которыми они описываются. Различают нелинейные, линейные, стационарные динамические системы, среди нелинейных систем дифференциальных уравнений выделяют системы с разрывными правыми частями. Вид дифференциальных уравнений определяет условия применимости теоремы существования и единственности решения, определяет сходимость и эффективность алгоритмов поиска решения в различных задачах оптимального управления.

В целом вид динамической системы (модели) определяется особенностями военно-технической системы, относительно которой принимается решение. Естественно стремление исследователя (лица, принимающего решение) к разработке наиболее простой модели - наиболее простым дифференциальным уравнениям, описывающим систему, именно в этом случае создается возможность построения эффективных алгоритмов поиска оптимального решения.

Итак, можно различать следующие основные виды дифференциальных уравнений, соответственно, динамических систем (моделей).

8.3.3. Ограничения вдоль траектории.

Дифференциальные уравнения описывают изменение состояния динамической системы при заданном управлении и в зависимости от начального и конечного состояния ограничивают возможные воздействия, прикладываемые к системе на интервале управления. Вместе с тем на состояния системы могут накладываться дополнительные ограничения, которые, в свою очередь, ограничивают возможные управления, более того сами управляющие воздействия должны удовлетворять, как правило, ряду ограничений.

Как отмечалось ранее, пространство альтернатив Δ представляет собой декартово произведение $\Delta = T \times X \times U$, где T - множество моментов времени; X - множество состояний системы; U - множество управлений. Поэтому ограничения, заданные на Δ, представляют собой конструкцию вида $G_t \times G_x \times G_u \subseteq \Delta$, причем, как правило, рассматриваются ситуации, когда $G_t = (t_o, t_f]$, в этом случае ограничения можно представить в виде $G_x(t) \times G_u(t) \subseteq \Delta$, $t \in (t_o, t_f]$. Следует отметить, что не зависимо от того, какие элементы пространства альтернатив подвергаются ограничениям (время, состояние или управление), обеспечить выполнение этих условий представляется возможным только на основе выбора соответствующих управлений.

Принято различать следующие способы задания ограничений вдоль траектории.

а). Ограничения на управление. Такого типа ограничения в общем виде можно представить как

$u(t) \in G_u = \{ u(t) \in R^m \mid g_i(t,u) \le 0, i \in I_u \}$.

Простейший вид таких ограничений, часто используемых в различных задачах оптимального управления, имеет вид

$|u(t)| \le U_o$.

б). Ограничения на фазовые переменные. Такие ограничения можно представить в виде

$x(t) \in G_x = \{ x(t) \in R^n \mid g_i(t,x) \le 0, i \in I_x \}$.

Например, движение динамической системы должно происходить в некоторой "трубке"

$|x(t)| \le a(t)$.

в). Совместные ограничения на управление и фазовые переменные. В ряде случаев ограничения на управление и ограничения на фазовые переменные не могут быть разделены. Тогда ограничения имеют вид

$G_{xu} = \{ x(t) \in R^n, u(t) \in R^m \mid g_i(t,x,u) \le 0, i \in I_{xu} \}$.

г). Интегральные ограничения (изопериметрические). Как функционал Лагранжа характеризует некоторое интегральное качество управления, так и изопериметрические ограничения описывают требования ко всему интервалу управления (ограничения количества используемого на управление ресурса, например)

$$\int_{t_0}^{t_f} G_i(t, x, u)\, dt = A_i, \; i = 1, \dots, z . \qquad (8.32)$$

8.3.4. Краевые условия.

Содержание задач оптимального управления, как правило, состоит в переводе динамической системы из некоторого начального состояния в конечное при достижении определенных качественных характеристик управления. В этой связи для постановки задачи весьма важны начальное и конечное состояния системы, которые задаются краевыми условиями.

Здесь, прежде всего, принято различать задачи с ф и к с и р о в а н н ы м временем (начальным $t_o = a$, или конечным $t_f = b$); и задачи со с в о б о д н ы м временем (начальным $t_o \in R_+$, или конечным $t_f \in R_+$).

Аналогично, с точки зрения задания ограничений на состояние в начальный и конечный момент времени различают задачи:

а). Задача с фиксированными концами
- фиксированный левый конец $x(t_o) = A \in R^n$,
- фиксированный правый конец $x(t_f) = B \in R^n$.

б). Задача со свободными концами
- свободный левый конец $x(t_o) \in R^n$,
- свободный правый конец $x(t_f) \in R^n$.

в). Задача с подвижными концами
- подвижный левый конец $E_{oi}(t_o,x(t_o)) = 0$, $i=1,..,l \le n$,
- подвижный правый конец $E_{fj}(t_f,x(t_f)) = 0$, $j=1,..,p \le n$.

В общем случае совокупность данных равенств определяет некоторую гиперповерхность на левом (правом) конце фазовой траектории. Если при задании такой гиперповерхности время не фиксировано, то говорят о задачах с "перемещающимся многообразием" на левом (правом) конце.

Анализ особенностей задания рассмотренных четырех элементов, участвующих в постановке задачи оптимального управления, позволяет провести классификацию различных задач и выбрать соответствующий метод решения.

8.4. Примеры задач оптимального управления

Рассмотрим несколько примеров простых задач оптимального управления, имеющих важные приложения.

8.4.1. Оптимальное управление сближением космических аппаратов

Два космических аппарата (КА) осуществляют совместное выполнение целевой задачи в процессе орбитального полета. Для инспекции технического состояния одного из космических аппаратов (второго), проводимой путем наружного осмотра, необходимо произвести их сближение. Для проведения сближения на борту первого КА имеется двигательная установка. Необходимо найти управление двигательной установкой КА, позволяющее произвести сближение за минимальное время.

Цель проводимой операции - сближение космических аппаратов, следовательно, их можно рассматривать, как систему, состоящую из двух материальных точек, перемещающихся в пространстве. Состояние такой системы можно описать векторной функцией времени $x(t) = \| x_1(t)\ x_2(t) \|^T$, где $x_1(t) = r(t)$ - скалярная функция времени, описывающая изменение относительной дальности между КА; $x_2(t) = v(t)$ - функция, описывающая изменение относительной скорости. Тогда дифференциальные связи компонент вектора состояния $x(t)$ описываются уравнением

$$\frac{d^2 r(t)}{dt^2} = u(t) ,$$

т.е. в соответствии со вторым законом Ньютона, ускорение равно прикладываемому импульсу. Иначе последнее уравнение можно переписать в виде системы двух дифференциальных уравнений

$$\frac{d}{dt} x_1(t) = x_2(t) .$$

$$\frac{d}{dt} x_2(t) = u(t) .$$

Прикладываемый импульс ограничивается техническими характеристиками двигательной установки, тогда

$$u(t) \in [-U_o, +U_o]$$

Начальное состояние системы соответствует некоторому положению КА в пространстве, тогда

$$x(t_o) = \| r(t_o)\ v(t_o) \|^T = \| r_o\ v_o \|^T .$$

Конечное состояние соответствует сближению КА, тогда

$$x(t_f) = \| 0\ 0 \|^T .$$

Произвести сближение следует за минимальное время, тогда функционал, описывающий качество управления имеет вид

$$\int_{t_0}^{t_f} dt \to \min.$$

Рассматриваемая задача является задачей оптимального управления с функционалом Лагранжа с фиксированными концами и свободным временем. Пусть $t_o = 0$, $t_f = T$, тогда задачу можно представить в виде

1. Функционал

$$\int_0^T dt \to \min.$$

2. Дифференциальные связи

$$\dot{x}_1 = x_2 ,$$
$$\dot{x}_2 = u .$$

3. Ограничения вдоль траектории

$u(t) \in [-U_o,+U_o]$ или $|u(t)| \le U_o$, $t \in (0,T]$.

4. Краевые условия
$x(0) = \| r_o \; v_o \|^T$; $x(T) = \| \; 0 \; 0 \; \|^T$.

8.4.2. Управление операциями обслуживания космического аппарата.

Для обеспечения целевого функционирования КА необходимо проводить ряд операций (работ), связанных с анализом состояния и управлением движения КА, связанных с анализом технического состояния и управлением бортовой аппаратурой. Такие операции проводятся с использованием наземных средств управления, расположенных на пунктах управления, в моменты времени, когда КА попадает в зону видимости того или иного пункта. В этих условиях необходимо найти программу управления выполнения операций обслуживания, позволяющую минимизировать расход ресурсов.

Целью управления является выполнение операций обслуживания, поэтому в качестве состояния системы x(t) целесообразно выбрать состояние операций $x(t) = \| x_1(t), x_2(t) ,\dots, x_n(t)\|^T$, где $x_i(t)$ - состояние i-ой операции обслуживания в момент времени t. Изменение состояния операций определяется дифференциальными уравнениями вида

$$\dot{x} = \varepsilon(t)\, d^T u(t) ,$$

где

$\varepsilon(t)$ - заданная функция, принимающая значение 1 в моменты времени, когда КА попадает в зону видимости того или иного пункта (операции обслуживания могут выполняться), и 0 в противном случае;

$d = \|d_1, d_2, \ldots, d_n\|^{т}$ - вектор, характеризующий максимально возможную интенсивность выполнения операций;

$u(t)$ - векторная функция времени, характеризующая управление выполнением операций обслуживания.

Начальное состояние операций $x(0) = 0$; конечное состояние $x(T)$, характеризующее факт выполнения операций, задано как $x(T) = x^{зад}$.

Необходимо найти программу выполнения операций обслуживания, минимизирующую расход ресурсов тогда функционал, описывающий качество управления имеет вид

$$\int_0^T u^{т}(t)\, R(t)\, u(t) dt \to \min.$$

где $R(t)$ - положительно определенная симметрическая матрица, характеризующая расходование ресурсов в момент времени t.

Тогда задача оптимального управления в целом имеет вид

1. Функционал

$$\int_0^T u^{т}(t)\, R(t)\, u(t) dt \to \min.$$

2. Дифференциальные связи

$$\dot{x} = \varepsilon(t)\, d^{т}\, u(t),$$

3. Ограничения вдоль траектории: $u(t) \in [0, 1]$; $x(t) \le x^{зад}$.

4. Краевые условия: $x(0) = 0$; $x(T) = x^{зад}$.

8.5. Качественный анализ проблемы оптимального управления

Поставленная задача оптимального управления динамической системой может быть решена с использованием некоторого алгоритма, но только в том случае, если решение (управление) такой задачи существует. Исследование принципиальной разрешимости задачи оптимального управления, доказательство существования и единственности решения и является предметом качественного анализа. Следует отметить, что существование оптимального управления в динамической системе (модели) не вытекает непосредственно из факта наличия управления в реальной системе. Последнее связано с тем обстоятельством, что поиск оптимального управления производится в рамках некоторого математического формализма - модели, которая является агрегированным (обобщенным) представлением исходной ситуации, и адекватность разработанной абстрактной модели исходной реальной ситуации и определяет существование решения в задаче оптимального управления.

Качественный анализ проблемы оптимального управления включает три основные группы вопросов.

1. Исследование принципиальной возможности перевода динамической системы из одного заданного состояния в другое за конечное время с использованием некоторого программного управления, принадлежащего заданному классу функций (управляемость динамической системы).

2. Анализ существования допустимого управления - управления, принадлежащего заданному классу функций, удовлетворяющего заданным ограничениям и переводящего динамическую систему из заданного начального состояния в заданное конечное состояние (достижимость состояний).

3. Анализ существования в классе допустимых управлений оптимального управления и его единственность.

8.5.1. Управляемость динамической системы.

Динамическая система $\dot{x}(t) = \varphi(x, u)$ называется управляемой [5, 43] в состоянии $x_1=x(t_1)$ относительно состояния $x_2=x(t_2)$, если существует кусочно-непрерывное управление $u(t)$, $t\in(t_1,t_2]$, позволяющее перевести систему из состояния x_1 в состояние x_2.

Если существует кусочно-непрерывное управление, позволяющее перевести динамическую систему из любого заданного состояния в любое желаемое состояние за конечное время, то система называется вполне (полностью) управляемой.

Для линейных стационарных систем (8.5), описываемых n дифференциальными уравнениями вида

$$\dot{x}(t) = Ax + Bu,$$

критерием полной управляемости является то, что ранг матрицы $G = \| B \mid AB \mid A^2B \mid \ldots \mid A^{n-1}B \|$ равен n, т.е.

$$\mathrm{rang}\| B \mid AB \mid A^2B \mid \ldots \mid A^{n-1}B \| = n\ . \tag{8.33}$$

Например, в задаче 1.4.1 оптимального управления сближением КА динамическая система описывается уравнениями

$$\dot{x}(t) = Ax + Bu = \begin{Vmatrix} 0 & 1 \\ 0 & 0 \end{Vmatrix} x + \begin{Vmatrix} 0 \\ 1 \end{Vmatrix} u ,$$

тогда критерий управляемости

$$\operatorname{rang} \| B \mid AB \| = \operatorname{rang} \begin{Vmatrix} 0 & 1 \\ 1 & 0 \end{Vmatrix} = 2 .$$

Таким образом, система вполне управляема.

Анализ управляемости нелинейных динамических систем достаточно сложен и здесь к настоящему времени получены весьма немногочисленные результаты.

Интересен подход [17] к анализу необходимых условий управляемости на основе исследования влияния управления на компоненты вектора состояния, которое описывается ориентированным графом. Действительно, так как управляемость это способность управления переводить систему из одного произвольного состояния в другое, то, очевидно, что если управление оказывает влияние на изменение всех компонент вектора состояния (прямое, или косвенное через другие компоненты), то система управляема. В ориентированном графе $\Gamma = (W,E)$ множество вершин W соответствует объединению $W = X \cup U$, где X - множество компонент вектора состояния X; U - множество компонент вектора управления; $E \subseteq W \times W$ - множество дуг, связывающих вершины из W, причем входные вершины соответствуют компонентам вектора состояния, стоящим в левой части дифференциальных уравнений, описывающих систему, а выходные вершины соответствуют компонентам вектора состояния и управления, стоящим в правой части дифференциальных уравнений. Для управляемости системы необходимо, чтобы все вершины X были достижимы (существовал путь) из вершин U.

В частности, в задаче 8.4.1 оптимального управления сближением КА множество вершин $W = \{x_1, x_2, u\}$, множество дуг $E = \{(x_2,x_1),(u,x_2)\}$ и из вершины (u) существует путь в вершину x_1, это путь $((u,x_2),(x_2,x_1))$, и существует путь в вершину x_2 - (u,x_2).

8.5.2. Достижимость состояний динамической системы

Вопрос достижимости заданного состояния динамической системы из некоторого начального состояния заключается в существовании допустимого управления - управления, принадлежащего заданному классу функций и удовлетворяющего заданным ограничениям.

В реальных задачах краевые условия, описывающие требуемое начальное и конечное состояние динамической системы, могут быть заданы так, что в силу ограничений на возможные управления (ограниченности ресурсов, например) система не в состоянии достигать конечного состояния, т.е. не может быть указано хотя бы одно допустимое управление. В этом случае решения задачи оптимального управления не существует. Действительно, в задаче 8.4.1 оптимального управления сближением КА конечное состояние $x(T) = (0,0)$ не может быть достигнуто из произвольного начального состояния, если допустимые управления $u(t)$ могут принимать только положительные значения, то есть $u(t)\in[0, +U_o]$, $t\in(0,T]$.

Анализ достижимости состояний динамической системы, с помощью которого оценивается выполнимость краевых условий, требует решения довольно нетривиальной задачи - построения областей достижимости или их аппроксимаций. В работах [39, 81] приводятся теоремы и алгоритмы, используемые для построения областей достижимости.

8.5.3. Существование и единственность оптимального управления

В задачах оптимального управления возможны случаи, когда при достижимости финального состояния динамической системы, т.е. в условиях существования допустимых управлений, оптимального управления не существует. Такие ситуации могут складываться, например, если функционал на множестве допустимых управлений может неограниченно убывать (принимать сколь угодно малые значения).

При качественном анализе процессов управления динамическими системами наряду с установлением факта существования оптимального управления целесообразно ставить вопрос о его единственности. В настоящее время отсутствуют универсальные критерии для оценки единственности решения в задачах оптимального управления динамическими системами, поэтому в каждой конкретной задаче следует, исходя из её специфики, проводить индивидуальное исследование данного вопроса.

8.6. Необходимые условия существования оптимального управления

При проведении исследований задач оптимального управления динамическими системами важную роль играют необходимые условия существования оптимального управления. Основное содержание этих условий заключается в формулировании свойств, присущих оптимальному управлению, т.е. свойств, которые имеют место, если оптимальное управление существует. С другой стороны, если такие условия (необходимые) не выполняются, то и оптимального управления не существует. В связи с последним положением необходимые условия довольно часто используются при качественном анализе существования оптимального управления. Однако прикладная роль этих условий значительно шире и помимо оценки качественных характеристик задачи, они позволяют выявить множество экстремальных управлений, т.е. подмножество допустимых управлений, содержащее оптимальное управление.

Исследование необходимых условий существования оптимального решения экстремальных задач, в которых решение описывается некоторой функцией, традиционно проводилось в рамках вариационного исчисления, созданного трудами выдающихся ученых прошлого таких, как Л.Эйлер, Ж.Лагранж, В.Гамильтон и другие. В 1956 г. группой российских ученых под руководством Л.С.Понтрягина были сформулированы в форме "принципа максимума" необходимые условия оптимальности для задач, в которых учитываются важные для практических приложений ограничения на возможные управления.

Принцип максимума существенно обобщает и развивает результаты классического вариационного исчисления и, по существу, образует фундамент всей современной теории оптимального управления. Именно его появление стимулировало дальнейшие исследования в области оптимального управления динамическими системами при решении практически важных задач. В результате проведения данных исследований произошла определенная стандартизация терминологии, методов и алгоритмов оптимального управления.

Рассмотрим принцип максимума Понтрягина для некоторых задач оптимального управления, при формулировании которого существенно используется некоторая специальная функция - функция Гамильтона.

8.6.1. Функция Гамильтона.

Изменение состояния $x(t) = (x_1(t),...,x_n(t))$ динамической системы под воздействием управления $u(t)$ описывается дифференциальными уравнениями (8.26)

$$\dot{x}(t) = \varphi(t, x, u),$$

Пусть необходимо найти такое управление $u(t)$, чтобы достигался минимум функционала (8.29) (задача Лагранжа)

$$J(x,u) = \int_{t_0}^{t_f} F(t,x,u)\,dt$$

Значение функционала $J(x,u)$ определяется, в первую очередь, управлением, поэтому такое значение можно рассматривать, как еще одну компоненту вектора состояния $x_o(t)=J(x,u)$. Тогда расширенный вектор состояния имеет $(n+1)$ компоненту

$$\bar{x}(t) = (x_o\ x_1, \ldots, x_n)^T. \tag{8.34}$$

Его изменение под воздействием управления описывается дифференциальными уравнениями

$$\frac{d\,x_i(t)}{dt} = \varphi_i(t, x, u),\ i=0,\ldots,n,$$

Причем

$$\varphi_o(t,x,u) = F(t,x,u). \tag{8.35}$$

Тогда вектор сопряженных переменных

$$\bar{\psi}(t) = (\psi_o(t), \psi_1(t), \ldots, \psi_n(t))^T$$

имеет $(n+1)$ компоненту и его изменение описывается системой дифференциальных уравнений (8.21)

$$\frac{d\psi_i(t)}{dt} = -\sum_{j=0}^{n} \frac{\partial \varphi_j(t,x,u)}{\partial x_i} \psi_j(t),\ i = 0,...,n, \tag{8.36}$$

Введем в рассмотрение ф у н к ц и ю Г а м и л ь т о н а, как скалярное произведение

$$H(t, \bar{x}, u, \bar{\psi}) = (\bar{\psi}, \varphi) = \varphi^T\,\bar{\psi} = \bar{\psi}^T \varphi = \sum_{j=0}^{n} \psi_j\,\varphi_j(t,x,u). \tag{8.37}$$

С использованием функции Гамильтона основную и сопряженную систему дифференциальных уравнений можно переписать в виде

$$\frac{d\,x_i(t)}{dt} = \frac{\partial H}{\partial \psi_i}, \; i = 0,\dots,n, \tag{8.38}$$

$$\frac{d\,\psi_i(t)}{dt} = \frac{\partial H}{\partial x_i}, \; i = 0,\dots,n, \tag{8.39}$$

Интегрируя основную и сопряженную системы дифференциальных уравнений, при известных начальных условиях, можно получать значения фазовых и сопряженных переменных в любой момент времени.

8.6.2. Принцип максимума Л.С.Понтрягина.

Рассмотрим формулировку принципа максимума для задачи поиска оптимального управления, переводящего динамическую систему из одного заданного состояния в другое заданное состояние, при этом управление должно в любой момент времени удовлетворять ограничениям и минимизировать некоторый функционал, отражающий интегральное качество управления. Такая задача является задачей Лагранжа со свободным временем и фиксированными концами.

$$J(x,u) = \int_{t_0}^{t_f} F(t,x,u)\,dt \to \min \tag{8.40}$$

$$\dot{x}(t) = \varphi\,(t, x, u), \tag{8.41}$$

$$u(t) \in G_u \subseteq R^m, \; t \in (t_o, t_f] \tag{8.42}$$

$$x(t_o) = x_o, \;\; x(t_f) = x_f. \tag{8.43}$$

Функция Гамильтона для задачи (8.40) - (8.43) будет иметь вид

$$H(t, \overline{x}, u, \overline{\psi}) = \psi_o F(t,x,u) + \psi^T \varphi\,(t,x,u). \tag{8.44}$$

Дифференциальные уравнения, описывающие изменение сопряженных переменных $\psi(t)$, имеют вид (8.36).

Необходимые условия оптимальности управления формулируются в виде следующего утверждения.

<u>Утверждение (Принцип максимума Л.С.Понтрягина).</u>

Если управление $u^*(t)$ является оптимальным управлением в задаче (8.40)-(8.43), то существует ненулевая непрерывная векторная функция $(\psi_o(t),\psi_1(t),...,\psi_n(t))$, удовлетворяющая сопряженной системе дифференциальных уравнений (8.36) и такая, что выполняются условия

1). Для любого момента времени $t \in (t_o, t_f]$ функция Гамильтона (8.44) достигает максимума по u(t), т.е.

$$H(t, \overline{x}, u^*, \overline{\psi}) = \max_{u \in G_u} H(t, \overline{x}, u, \overline{\psi}). \tag{8.45}$$

2). Выполняются соотношения

$$\psi_o(t_f) \le 0; \; H(t_f, \overline{x}, u^*, \overline{\psi}) = 0. \tag{8.46}$$

Замечание 1.

Условие 1 принципа максимума может служить основой для вычисления оптимального управления на основе решения некоторой специализированной краевой задачи. Действительно, изменение фазовых переменных x(t) описывается дифференциальными уравнениями (8.41), изменение сопряженных переменных $\psi(t)$ описывается дифференциальными уравнениями (8.36), при известных начальных условиях (x_o, ψ_o) фазовые и сопряженные переменные для любого момента времени t можно вычислить. В этом случае гамильтониан в (8.45), становится функцией одного переменного u, которое может быть вычислено для момента времени t при соблюдении ограничений $u \in G_u$.

Последовательно решая задачу Коши для (2n+2) дифференциальных уравнений (8.41), (8.36), описывающих изменение фазовых и сопряженных переменных, для каждого момента времени $t \in (t_o, t_f]$, и, находя оптимальное управление из условия максимизации гамильтониана, можно найти оптимальную программу управления u*(t) в целом. Такая задача называется к р а е в о й задачей.

Основная особенность рассматриваемой здесь краевой задачи заключается в том, что начальные условия для сопряженных переменных $\psi(t_o)$ не заданы. Вместе с тем фазовая траектория должна быть такова, чтобы $x(t_f)=x_f$, и в момент t_f выполнялись соотношения (8.46).

Замечание 2.

Формулировки принципа максимума для различных задач оптимального управления отличаются видом функции Гамильтона (т.к. могут использоваться разные функционалы) и условиями 2 в формулировке принципа максимума. Например, в рассмотренной задаче Лагранжа (8.40)-(8.43), но с фиксированным временем и свободным правым концом условия 2 принципа максимума будут иметь вид

$$2'). \ \psi_o(t_f) < 0; \ \ \psi_i(t_f) = 0, \ i=1,...,n. \qquad (8.47)$$

Такие условия (условия 2) довольно часто называются условиями т р а н с в е р с а л ь н о с т и. В силу особой важности данных условий для задач оптимального управления следует остановиться на них несколько более подробно.

8.6.3. Условия трасверсальности.

В общем случае состояние системы в конечный (или начальный) момент времени должно принадлежать некоторой области E_f (или E_o). Рассмотренные случаи задания условий трансверсальности в виде (8.46) и (8.47) соответствуют предельным ситуациям, когда правый конец фазовой траектории фиксирован или свободен. Рассмотрим для правого конца более общую ситуацию, когда $x(t_f)\in E_f$ (аналогичную ситуацию можно рассмотреть и для левого конца траектории, когда $x(t_o)\in E_o$).

Поскольку состояние системы не фиксировано ($x(t_f)\in E_f$), то условия трансверсальности позволяют сформировать дополнительные ограничения для однозначного решения задачи оптимального управления и нахождения оптимального состояния системы в конечный момент времени. По существу, выполнение условий трансверсальности означает то, что вектор сопряженных переменных в момент времени t_f ($\psi(t_f)$) ортогонален гиперплоскости, касательной к области E_f в точке $x(t_f)$.

Пусть область E_f задается совокупностью ограничений

$$E_{fj}(x(t_f)) = f_j(x_f)= 0, \ \ j = 1,..., p \le n. \qquad (8.48)$$

Данные ограничения задают некоторую гиперповерхность. Точки x, находящиеся в плоскости, касательной к j-ой гиперповерхности, определяются равенством

$$\mathrm{grad}^T f_j(x_f)\ (x - x_f) = 0, \ \ j = 1,...,p,$$

или

$$(f_{jx}(x_f), \ x - x_f) = 0, \ \ j = 1,....,p. \qquad (8.49)$$

Здесь

$f_{jx}(x_f)$, - вектор частных производных функции f_j по компонентам вектора состояния x, вычисленный в точке $x_f=x(t_f)$;

$$f_{jx}(x_f) = \left\| \frac{\partial f_j(x_f)}{\partial x_1} \; \frac{\partial f_j(x_f)}{\partial x_2} \; \ldots\ldots \; \frac{\partial f_j(x_f)}{\partial x_n} \right\|, \; i = 1,\ldots, n .$$

Ортогональность вектора $\psi(t_f)$ к некоторой гиперплоскости Г в точке x_f означает, что

$$(\psi(t_f) , x - x_f) = 0, \text{ где } x \in Г. \qquad (8.50)$$

Тогда из совместного рассмотрения (8.49), (1,50) следует, что существуют числа $a_1,...,a_p$, такие, что

$$\psi(t_f) = \sum_{j=1}^{p} a_j f_{jx}(x_f) . \qquad (8.51)$$

Условия (8.51) совместно с условиями (8.48) $f_j(x_f) = 0$, $j = 1,...,p$ задают (n+p) ограничений для (n+p) неизвестных ($\psi_i(t_f)$, $i = 1 ,\ldots,n$, и a_j, $j=1,....,p$). Эти условия и являются условиями трансверсальности для задачи с подвижным правым концом.

8.7. Многошаговые процессы управления рекуррентными динамическими системами

Решение задач оптимального управления основывается на использовании некоторых вычислительных процедур и предполагает, как правило, при проведении расчетов использование вычислительной техники (электронных вычислительных машин). При этом состояние динамической системы и управление оценивается в дискретные моменты времени, связанные с шагом численного интегрирования дифференциальных уравнений. Наряду с этой особенностью численного решения задач оптимального управления, в ряде случаев сама природа процессов, протекающих в системе, такова, что состояние системы изменяется в дискретные моменты времени. В этой связи возникает необходимость проведения анализа того, в какой степени результаты теории оптимального управления, полученные при исследовании непрерывных систем, можно распространить на системы с дискретным временем.

Динамические системы, состояние которых изменяется в дискретные моменты времени, называются р е к у р р е н т н ы м и динамическими системами или системами с дискретным временим. Процессы изменения состояния и управления в таких системах называются м н о г о ш а г о в ы м и процессами.

Рекуррентные динамические системы.

Изменение состояния непрерывной автономной динамической системы описывается системой дифференциальных уравнений вида

$$\dot{x}(t) = \varphi(x, u), \ t \in (t_o, t_f]. \tag{8.52}$$

Пусть теперь состояние изменяется в дискретные моменты времени, принимающие значения из множества T

$$T = \{t_o, t_1, t_2, ..., t_N\}, \text{ где } t_N = t_f.$$

Тогда моменту времени t_k можно сопоставить его номер k и изменение состояния описать рекуррентной системой уравнений

$$x(k+1) = f(x(k), u(k)), \ k \in \{0, ..., N-1\}. \tag{8.53}$$

Учитывая, что состояние динамической системы меняется в дискретные моменты времени, уравнения (8.52) можно переписать в виде

$$\frac{x(k+1) - x(k)}{(k+1) - k} = \varphi(x, u). \tag{8.54}$$

Откуда изменение состояния на k-м шаге описывается конечно-разностными уравнениями

$$x(k+1) = x(k) + \varphi(x(k), u(k)), \ k \in \{0, ..., N-1\}, \tag{8.55}$$

которые описывают динамический объект близкий (но не тождественный) к объекту (8.53).

Задача оптимального управления (8.25)-(8.28), поставленная для непрерывных систем, может быть переписана для рекуррентных систем

$$J(x,u) = \sum_{k=0}^{N-1} F(x(k), u(k)) \to \min \tag{8.56}$$

$$x(k+1) = f(x(k), u(k)), \ k = 0, ..., N-1, \tag{8.57}$$

$$u(k) \in G_u \subseteq R^m, \ k = 0, ..., N-1, \tag{8.58}$$

$$x(0) = x_o, \ x(N) = x_f. \tag{8.59}$$

Формулирование необходимых условий оптимальности решения, аналогичных принципу максимума Понтрягина, с учетом проведенной замены производной отношением приращений (8.54), для задачи (8.56)-(8.59) в общем случае неверно. Поэтому для справедливости принципа максимума необходимо накладывать дополнительные ограничения на управляемый объект.

Для управляемого объекта (8.53), как и для непрерывных систем можно записать уравнения, описывающие изменение сопряженных переменных

$$\psi_i(k+1) = -\sum_{j=1}^{n} \frac{\partial \varphi_j(x(k),u(k))}{\partial x_i}\psi_j(k),\ i = 0,1,\ldots,n,$$

Тогда функция Гамильтона будет иметь вид

$$H(\bar{x}(k),u(k),\bar{\psi}(k)) = \psi_o F(x(k),u(k)) + \psi^T f(x(k),u(k)).$$

Рассмотрим особенности формулирования принципа максимума Л.С.Понтрягина для рекуррентных систем.

Дискретный принцип максимума.

В задачах оптимального непрерывного управления решение в любой момент времени находится на основе поиска максимума функции Гамильтона. В задачах оптимизации управления рекуррентными динамическими системами можно показать [63], что функция Гамильтона вдоль оптимальной траектории отличается от своего максимального значения на величину порядка $O(\tau)$, где $\tau \geq \max(t_i - t_j), \forall_i, j \in \{0,\ldots,N\}$. Таким образом, при увеличении шага дискретизации значение функции Гамильтона все больше отличается от своего максимального значения, и естественно предполагать, что для произвольных разностных уравнений принцип максимума вообще не имеет места. В этой связи на управляемый объект необходимо накладывать дополнительные ограничения, в качестве таких ограничений выступает следующее условие.

Условие 1. Пусть множества достижимости рекуррентной динамической системы (8.53) за один шаг $R(x(k))$ выпуклы при любом $x(k) \in R^n$. Здесь $R(x(k))$ определяется как $R(x(k)) = \{x(k+1) \mid x(k+1)=f(x(k),u(k)),\ \forall u(k) \in G_u,\ k=0,..,N-1\}$.

Теперь приведем дискретный принцип максимума [76].

Утверждение.

Пусть в задаче оптимального управления рекуррентной динамической системой (8.56)-(8.59) управляемый объект (8.57) удовлетворяет условию 1. Тогда для существования оптимального управления $u^*(k)$), $k=0,\ldots,N-1$, необходимо существование такой тождественно не равной нулю функции ($\psi_o(k)$, $\psi_1(k),\ldots,\ \psi_n(k)$), что функция Гамильтона принимает максимальное значение по управлению на любом шаге k, т.е.

$$H\left(\overline{x}(k), u^*(k), \overline{\psi}(k)\right) = \max_{u(k)\in G_u} \left(\psi_o F\left(x(k), u(k)\right) + H\left(x(k), u(k), \psi(k)\right)\right).$$

Из данного утверждения следует, что в дискретном принципе максимума появляется дополнительное условие 1. Для ряда динамических систем, в частности, для систем вида

$x(k+1) = f(x(k)) + B(x(k))\ u(k)$

проверку условия 1 можно заменить проверкой выпуклости множества G_u [42], хотя в общем случае проверка данного условия довольно затруднительна.

Контрольные вопросы

1. В какой форме может описываться решение в задачах оптимизации, использующих динамические модели ?
2. Какое новое понятие, помимо понятия «решение», необходимо вводить при построении динамических моделей принятия решений ?
3. Как называются алгоритмы позволяющие найти оптимальное решение на основе использования динамических моделей ?
4. Каким образом найти состояние динамической системы в любой момент времени; что для этого необходимо знать; как называется соответствующая задача ?
5. Какие качественные характеристики задач оптимального управления различают; для чего необходимо проведение качественных исследований ?
6. Из каких элементов состоит постановка задачи оптимального управления; каким образом в этих задачах задается множество допустимых альтернатив, что является аналогом целевой функции ?
7. В чем суть принципа максимума, как необходимого условия оптимальности управления, при каких условиях этот принцип задает достаточные условия оптимальности ?
8. Каким образом на основе принципа максимума найти оптимальное управление? Что для этого необходимо сделать в первую очередь, во вторую …?
9. Чем отличаются формулировки принципа максимума для различных задач оптимального управления ?
10. Чем определяется роль многошаговых задач в практике оптимального управления космическими средствами ?

9. АЛГОРИТМЫ ВЫБОРА ОПТИМАЛЬНОГО ПРОГРАММНОГО УПРАВЛЕНИЯ ДИНАМИЧЕСКИМИ СИСТЕМАМИ

9.1. Вычислительные методы расчета оптимальных программ

Постановкой задачи оптимального управления заканчивается этап формализации принятия решения, т.е. этап построения формальной математической модели исходной ситуации принятия решения. В случаях когда, исходя из специфики исходной ситуации, решение должно описываться функцией (в общем случае векторной функцией) времени (функцией состояния или выхода), эта модель является динамической. Далее возникает проблема разработки алгоритма (формальной последовательности действий), позволяющего найти наилучшее решение. Такие алгоритмы базируются на некоторых математических методах поиска оптимальных решений. В данной главе будем рассматривать методы (а, соответственно, и алгоритмы) поиска решения в виде векторной функции времени. Такое решение часто называют оптимальной программой управления (планом), а соответствующие задачи называют задачами поиска оптимального программного управления.

Эффективность различных вычислительных методов и алгоритмов оптимального управления, реализующих методы поиска оптимального решения, определяется, прежде всего, временем счета и потребным объемом памяти электронной вычислительной машины, необходимого для хранения промежуточных данных вычислений. Так как все алгоритмы строятся, как правило, по итерационной схеме, то время счета определяется с одной стороны скоростью сходимости алгоритма (количеством итераций, которые необходимо провести для получения приемлемой точности результатов) и, с другой стороны, трудоемкостью вычислений на отдельной итерации. Большое значение приобретают такие свойства алгоритмов, как возможность стандартизации вычислений и возможность построения универсальных алгоритмов, позволяющих решать достаточно широкий класс задач оптимального управления.

Различные вычислительные методы оптимального управления принято разделять на два больших класса: прямые методы и непрямые методы.

9.1.1. Прямые методы.

Под прямыми методами решения задач оптимального управления в настоящее время принято понимать методы, не использующие непосредственно необходимых или достаточных условий оптимальности.

Прямые методы можно разделить на три группы:

1). Методы, основанные на редукции исходной задачи оптимального управления к некоторой задаче математического программирования, т.е. задаче поиска экстремума функции в конечномерном пространстве. В этом случае дифференциальные уравнения заменяются конечноразностными уравнениями (см. п. 8.7). Получающиеся при этом задачи имеют специфические особенности, позволяющие развивать вычислительные процедуры, обладающие более высокой эффективностью, чем стандартные алгоритмы математического программирования.

2). Методы, основанные на использовании градиентных методов в функциональных пространствах.

3). Методы случайного поиска.

В методах первой, второй и третьей групп различают поиск оптимального решения в пространстве управлений и поиск решения в пространстве состояний (фазовом пространстве).

9.1.2. Непрямые методы.

Непрямые методы основаны на использовании необходимых или достаточных условий оптимальности.

Наиболее развитыми в алгоритмическом смысле здесь являются методы, базирующиеся на необходимых условиях оптимальности, которые могут быть получены на основе вариационного исчисления или на основе принципа максимума Л.С. Понтрягина. Использование необходимых условий оптимальности позволяет свести задачу поиска минимума целевой функции к задаче отыскания корней некоторой другой функции, а исходную задачу оптимального управления свести к некоторой специальной краевой задаче для системы обыкновенных дифференциальных уравнений.

Использование достаточных условий оптимальности для нахождения оптимального управления встречает значительные трудности теоретического характера, связанные, прежде всего, с корректным формулированием таких условий. Результаты исследований, проведенных в данном направлении, позволяют говорить об эффективности подхода только в некоторых специальных приложениях.

9.2. Методы, основанные на решении краевой задачи

Необходимые условия в форме принципа максимума Л.С. Понтрягина могут служить основой для сведения задачи оптимального управления к некоторой специализированной краевой задачи.

Рассмотрим особенности применения данного подхода на примере задачи Лагранжа со свободным временем и фиксированными концами.

$$J(x,u) = \int_{t_0}^{t_f} F(t,x,u)\,dt \rightarrow \min \tag{9.1}$$

$$\dot{x}(t) = \varphi\,(t, x, u), \tag{9.2}$$

$$u(t) \in G_u \subseteq R^m,\ t \in (t_o, t_f] \tag{9.3}$$

$$x(t_o) = x_o,\ \ x(t_f) = x_f. \tag{9.4}$$

При известных дифференциальных уравнениях (9.2), описывающих изменение во времени фазовых переменных, можно записать дифференциальные уравнения для сопряженных переменных

$$\frac{d\psi_i(t)}{dt} = -\sum_{j=0}^{n} \frac{\partial \varphi_j(t,x,u)}{\partial x_i}\psi_j(t),\ i = 0,\dots,n, \tag{9.5}$$

Функция Гамильтона для задачи (9.1)-(9.4) будет иметь вид

$$H(t, \overline{x}, u, \overline{\psi}) = \psi_o F(t,x,u) + \psi^T \varphi\,(t,x,u). \tag{9.6}$$

В соответствии с принципом максимума для того, чтобы $u^*(t)$ являлось оптимальным управлением в задаче (9.1)-(9.4), необходимо существование ненулевой непрерывной векторной функции сопряженных переменных $(\psi_o(t),\psi_1(t),\dots,\psi_n(t))$ такой, что для любого момента времени $t\in(t_o,t_f]$ функция Гамильтона (9.6) достигает максимума по $u(t)$, т.е.

$$H\left(t,\overline{x},u^*,\overline{\psi}\right)=\max_{u\in G_u}\left(H\left(t,\overline{x},u,\overline{\psi}\right)\right),\ \forall t\in(t_o,t_f]\ . \qquad (9.7)$$

Тогда при известных начальных условиях $(x_o,\ \psi_o)$ в момент t_o функция Гамильтона является функцией одного векторного аргумента $u(t_o)$. В соответствии с (9.7) $u^*(t_o)$ находится в результате решения задачи математического программирования, существо которой состоит в нахождении максимума функции Гамильтона при $u\in G_u$.

Далее, при известных (x_o,ψ_o) и управлении $u^*(t_o)$, на основе решения задачи Коши для систем дифференциальных уравнений (9.2), (9.5) можно вычислить $(x(t_1),\psi(t_1))$ в момент времени t_1, $t_o < t_1 \le t_f$. Тогда управление $u^*(t_1)$ снова находится, исходя из максимизации функции Гамильтона в соответствии с (9.7).

Таким же образом можно вычислить управление для момента времени t_2, такого, что $t_1 <, t_2 \le t_f$, и так далее...

Итак, последовательно решая задачу Коши для 2n дифференциальных уравнений (9.2), (9.5), описывающих изменение фазовых и сопряженных переменных, в каждый момент времени $t\in(t_o,t_f]$, и, находя оптимальное управление из условия максимизации гамильтониана, можно найти оптимальную программу управления $u^*(t)$ в целом.

Задача нахождения управления, позволяющего перевести динамическую систему, описываемую дифференциальными уравнениями (9.2), (9.5), из начального состояния (x_o,ψ_o) в конечное состояние (x_f,ψ_f), называется к р а е в о й задачей.

Основная особенность данной задачи состоит в том, что, если начальные условия для фазовых переменных заданы $(x(t_o)=x_o)$, то начальные условия для сопряженных переменных $\psi(t_o)$ не заданы. Вместе с тем, фазовая траектория должна быть такова, чтобы выполнялись краевые условия на правом конце $x(t_f)=x_f$. В этой связи, если задать произвольные значения $\psi(t_o)=\alpha$, то, решая краевую задачу для начального состояния (x_o,α), найдем некоторое управление, переводящее систему (9.2) в некоторое состояние $x(t_f,\alpha)$, которое в общем случае может быть отлично от заданного состояния x_f.

Невязки $(x(t_f,\alpha) - x_f)$, естественно, являются функциями начальных значений сопряженных переменных α. Обозначим такие невязки, как $d(\alpha)$

$$d(\alpha) = x(t_f,\alpha) - x_f. \tag{9.8}$$

Тогда для решения поставленной задачи нахождения оптимального управления, переводящего динамическую систему и состояния x_o в состояние x_f, необходимо найти значения α, которые обращают невязки $d(\alpha)$ в нули.

Таким образом, исходная задача оптимального управления на основе принципа максимума сведена к задаче нахождения α, при которых $d(\alpha) = 0$, т.е. к задаче нахождения корней α трансцендентного уравнения

$$d(\alpha) = x(t_f,\alpha) - x_f = 0. \tag{9.9}$$

Существенной особенностью данной задачи является то, что векторная функция $d(\alpha)$ задана посредством алгоритма, который связан с необходимостью интегрирования системы 2n дифференциальных уравнений (9.2),(9.5) при начальных условиях (x_o,α), причем в процессе интегрирования управление $u(t)$ находится, исходя из максимизации функции Гамильтона на основе решения вспомогательной задачи математического программирования.

9.3. Метод Ньютона

Метод Ньютона является исторически первым численным способом решения трансцендентного уравнения (9.9), который предназначен для поиска оптимального управления. Суть метода, носящего итеративный характер, состоит в уточнении значений α, проводимого с целью последовательного сокращения невязок $d(\alpha)$. Тогда на некоторой k-ой итерации

$$\alpha^{k+1} = \alpha^k + \delta^k, \quad k = 0,1,\ldots. \tag{9.10}$$

Здесь δ^k - вектор поправки к значениям сопряженных переменных на k-ой итерации.

Значение δ^k было бы естественно выбирать из условия минимизации невязки $d(\alpha^{k+1})$. Пусть $k = 0$, тогда, полагая δ^o малыми, и значения $d(\alpha^1)$ дифференцируемыми в окрестности α^o, для i-ой компоненты $d_i(\alpha^1)$ можно записать

$$d_i(\alpha^1) = d_i(\alpha^o) + \nabla^т d_i(\alpha^o)\,(\alpha^1 - \alpha^o)\,,$$

или с учетом (9.10)

$$d_i(\alpha^1) = d_i(\alpha^o) + \nabla^т d_i(\alpha^o)\,\delta^o\,, \qquad (9.11)$$

Здесь $\nabla^т d_i(\alpha^o)$ - транспонированный вектор градиента невязки d_i по компонентам α

$$\nabla^т d_i\left(\alpha^o\right) = \left(\frac{\partial d_i}{\partial \alpha_1}\left(\alpha^o\right), \frac{\partial d_i}{\partial \alpha_2}\left(\alpha^o\right), \ldots, \frac{\partial d_i}{\partial \alpha_n}\left(\alpha^o\right)\right)^т$$

Иначе (9.11) можно переписать

$$d_i\left(\alpha^1\right) = d_i\left(\alpha^o\right) + \sum_{j=1}^{n} \frac{\partial d_i}{\partial \alpha_j}\left(\alpha^o\right)\delta_j^o,\ i = 1,\ldots,n\,. \qquad (9.12)$$

В целом, исходя из (9.11), вектор невязок определяется как

$$d(\alpha^1) = d(\alpha^o) + D(\alpha^o)\,\delta^o, \qquad (9.13)$$

где $D(\alpha^o)$ - матрица Якоби, вычисленная в α^o,

$$D\left(\alpha^o\right) = \left\| \frac{\partial d_i}{\partial \alpha_j}\left(\alpha^o\right) \right\|$$

Значения α^o следует выбирать таким образом, чтобы невязки $d(\alpha^1)$ в (9.13) свести к нулю. Тогда

$$\delta^o = -D^{-1}(\alpha^o)\, d(\alpha^o). \qquad (9.14)$$

В целом схема итерационного процесса в соответствии с методом Ньютона с учетом (9.10), 2(14) будет иметь вид

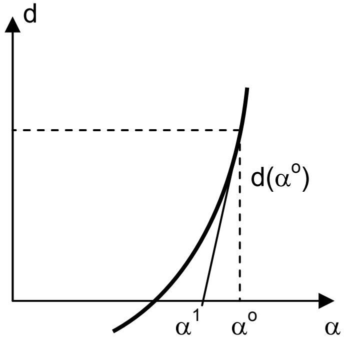
$$\left.\begin{aligned} \delta^k &= -D^{-1}(\alpha^k)\, d(\alpha^k), \\ \alpha^{k+1} &= \alpha^k + \delta^k, \end{aligned}\right. \qquad k = 0,1,\ldots. \qquad (9.15)$$

На каждой итерации производится вычисление матрицы $D(\alpha^k)$, для чего необходимо решать (n+1) задачу Коши для системы 2n уравнений (9.2), (9.5). При этом используются приближенные конечно-разностные формулы вида

$$\frac{\partial d_i}{\partial \alpha_j}\left(\alpha^k\right) \approx \frac{d_i\left(\alpha_1^k, \ldots, \alpha_j^k + \sigma_j, \ldots, \alpha_n^k\right) - d_i\left(\alpha_1^k, \ldots, \alpha_j^k, \ldots, \alpha_n^k\right)}{\sigma_j}$$

В соответствии с последней формулой один раз рассчитываются невязки $d(\alpha^k)$, i=1,...,n, при этом решается задача Коши для 2n дифференциальных уравнений. Затем решается n задач Коши, в которых придается поочередно приращение σ_j всем компонентам α^k_j j=1,...,n.

Метод Ньютона, предназначенный для поиска корней функции, часто называют методом касательных. Действительно, рассмотрим одномерный случай поиска корня функции $d(\alpha)$, тогда ситуацию можно характеризовать следующим рисунком



$$D(\alpha^o) = \dot{d}(\alpha^o) = \frac{d(\alpha^o)}{\delta^0}$$

$$\delta^o = -d(\alpha^o)/\dot{d}(\alpha^o)$$

$$\alpha^1 = \alpha^o + \delta^o$$

$$D(\alpha^1) = \dot{d}(\alpha^1)$$

....

В случаях, когда начальное приближение α^o выбирается достаточно близко к корню функции, метод Ньютона сходится достаточно быстро. Вместе с тем, при произвольном выборе α^o метод Ньютона может расходиться, что является его существенным недостатком.

Существует большое число модификаций метода Ньютона, позволяющих преодолеть указанный недостаток. В частности, из выражения (9.15) следует, что

$\alpha^1 = \alpha^o + \delta^o$,

тогда, если

$$\|d(\alpha^1)\| \geq \|d(\alpha^o)\|, \qquad (9.16)$$

то следует уменьшить поправку δ^o (обычно δ^o домножают на $\beta \in (0, 1)$). При новом значении $\alpha^1 = \alpha^o + \beta\delta^o$ проверяют выполнение условия (9.16) и, если оно выполняется , то производится дальнейшее уменьшение β (как правило, β = 1/2, 1/4, 1/8,...). Подобная модификация метода не гарантирует его сходимости, но расширяет область применения метода.

Алгоритм Ньютона

0). Исходное состояние. Задается начальное приближение значений сопряженных переменных α^o; критерий окончания процесса ε; номер шага k = 0.

1). Решается задача Коши для системы 2n дифференциальных уравнений вида (9.2), (9.5) при начальных условиях (x_o,α^o). На каждом шаге численного интегрирования управление u(t) находится из условия максимизации функции Гамильтона в соответствии с условием

$$H\left(t, \bar{x}, u^*, \bar{\psi}\right) = \max_{u \in G_u} \left(H\left(t, \bar{x}, u, \bar{\psi}\right)\right), \ \forall t \in (t_o, t_f] \ .$$

2). Вычисляется $d(\alpha^k) = x(t_f,\alpha^k) - x_f$.
Если $\| d(\alpha^k) \| \le \varepsilon$, то процесс счета окончен.
Если $\| d(\alpha^k) \| > \varepsilon$, то производится переход на шаг 3.

3). Решается n задач Коши, аналогично задаче п.1, но при начальных условиях $\alpha^k(\sigma_j)$, j=1,...,n. Здесь $\alpha^k(\sigma_j)$ - вектор, компоненты которого определяются по правилу

$$\alpha_i^k\left(\sigma_j\right) = \begin{cases} \alpha_i^k \quad , & i \ne j \\ \alpha_i^k + \sigma_j, & i = j \end{cases}$$

В результате получаем n значений векторов невязок

$$d(\alpha^k(\sigma_j)) = x(t_f, \alpha^k(\sigma_j)) - x_f, \ \ j = 1,...,n.$$

4). Строится матрица $D(\alpha^k)$

$$D\left(\alpha^k\right) = \left\| \frac{d_i\left(\alpha^k\left(\sigma_j\right)\right) - d_i\left(\alpha^k\right)}{\sigma_j} \right\|$$

5). Производится вычисление поправки δ^k и нового значения вектора сопряженных переменных α^{k+1}

$$\delta^k = -D^{-1}(\alpha^k)\, d(\alpha^k),$$
$$\alpha^{k+1} = \alpha^k + \delta^k,$$

6). k = k+1. Переход на шаг 1.

Описанный алгоритм Ньютона широко используется при решении разнообразных задач оптимального управления. Основные трудности его использования связаны, как отмечалось, с неопределенностью задания начального приближения сопряженных переменных α^o. Другие трудности использования рассмотренного алгоритма Ньютона связаны с низкой устойчивостью решения, которая проявляется при больших интервалах управления $(t_o,t_f]$. В этом случае незначительные изменения начальных значений сопряженных переменных α могут вызывать значительные изменения решения $x(t_f,\alpha)$. По этим причинам метод Ньютона, несмотря на его простоту и удобство описания, не стал универсальным методом расчета оптимальных управлений на основе решения краевых задач.

В этой связи стали развиваться другие способы решения краевых задач, в частности, методы переноса граничных условий, методы, использующие процедуры решения задач со свободным правым концом [52, 53, 63].

9.4. Алгоритм И.Крылова-Ф.Черноусько

Метод, предложенный И.А. Крыловым и Ф.Л. Черноусько в 1962 году [53], первоначально использовался для решения задач оптимального управления со свободным правым концом. В то же время, вводя в функционал терминальный член, штрафующий отклонение фазовых переменных в финальный момент времени от заданных значений, данный метод может быть распространен на рассматриваемые задачи с фиксированными концами.

Основная идея метода состоит в том, что задается (каким либо образом) некоторое д и с п е т ч е р с к о е управление $u^o(t)$. Тогда при известных начальном состоянии x_o и управлении $u^o(t)$ можно вычислить (решая задачу Коши) состояние системы в финальный момент времени, которое зависит от диспетчерского управления, $x(t_f,u^o)$. Такое состояние может не совпадать с заданным состоянием x_f, но можно вычислить значение сопряженных переменных в финальный момент $\psi(t_f)$. Тогда, интегрируя сопряженную систему дифференциальных уравнений в о б р а т н о м времени, можно получить значение сопряженных переменных в начальный момент времени $\psi(t_o)$. Теперь на левом конце траектории имеются начальные условия для фазовых и сопряженных переменных $x(t_o)$ и $\psi(t_o)$. Тогда, интегрируя основную и сопряженную систему дифференциальных уравнений в прямом времени, и вычисляя управляющие воздействия, исходя из максимизации функции Гамильтона, находится новое управление $u^1(t)$, которое заменяет диспетчерское. Процедура продолжается до тех пор, пока $\|x(t_f,u^k) - x_f\| > \varepsilon$, k=0,1,2,…

Рассмотрим соответствующий алгоритм.

Алгоритм.

0). Исходное состояние. На основе анализа исходной задачи задается диспетчерское управление $u^o(t)$; критерий окончания процесса ε; номер шага $k = 0$;

1). Решается задача Коши для системы n дифференциальных уравнений (9.2)

$$\dot{x}(t) = \varphi(t, x, u^k),\ t \in (t_o, t_f]$$

при известных начальных условиях $x(t_o) = x_o$ и диспетчерском управлении $u^k(t)$. В результате находим состояние системы в финальный момент времени $x(t_f, u^k)$.

2). Рассчитываются значения сопряженных переменных в финальный момент времени по формулам

$$\psi_i(t_f) = -S_i \frac{\partial \left(x_i\left(t_f, u^k\right) - x_{fi}\right)^2}{\partial x_i}, \quad i = 0, 1, \ldots, n, \tag{9.17}$$

где S_i - константа, характеризующая размер штрафа за нарушение i-го краевого условия.

3). Производится интегрирование сопряженной системы дифференциальных уравнений в обратном направлении времени (от момента t_f к моменту времени t_o)

$$\dot{\psi}(t) = \xi(t, \psi),\ t \in (t_f, t_o]$$

при начальных условиях $\psi(t_f)$. В результате получается траектория изменения сопряженных переменных, и в момент времени t_o вычисляется значение $\psi(t_o)$.

4). При известных начальных условиях $x(t_o)$, $\psi(t_o)$ решается система 2n дифференциальных уравнений

$$\dot{x}(t) = \varphi(t, x, u^k),\ t \in (t_o, t_f]$$

$$\dot{\psi}(t) = \xi(t, \psi),\quad t \in (t_o, t_f]$$

На каждом шаге ищется управление $u^{k+1}(t)$, исходя из условия максимизации функции Гамильтона

$$H(t,\overline{x},u^{k+1},\overline{\psi}) = \max_{v \in G_u} H(t,\overline{x},v,\overline{\psi}).$$

5). Проверяется критерий окончания итерационного процесса: если $\| x(t_f, u^{k+1}) - x_f \| \le \varepsilon$, то процесс заканчивается, если $\| x(t_f, u^{k+1}) - x_f \| > \varepsilon$, тогда k = k+1, и производится переход на шаг 2.

Рассмотренный алгоритм показал достаточную эффективность при решении ряда задач оптимального управления космическими средствами. Вместе с тем в ряде случаев он может расходиться. В этой связи был предложен ряд способов улучшения сходимости. В частности, один из этих способов заключается в том, что на k-ой итерации новое приближение управления u^{k+1}, которое вычисляется в соответствии с п.4, реализуется не на всем интервале $(t_o, t_f]$, а лишь на его части $(t^k, t_f]$, где $t^k \in (t_o, t_f]$. Тогда на k-ой итерации управление $u^{k+1}(t)$ имеет вид

$$u^{k+1}(t) = \begin{cases} u^k(t) & , \quad t \in (t_o, t^k] \\ u^{k+1}(t) & , \quad t \in (t^k, t_f] \end{cases}$$

Здесь $u^k(t)$ - управление предыдущего шага,

$u^{k+1}(t)$ - управление, вычисляемое в соответствии с п.4 алгоритма.

Действительно, если $t^k = t_f$, то управление, а, соответственно, и значение функционала на k-ой и (k+1)-ой итерации совпадают. С другой стороны, если $t^k = t_o$, то естественно ожидать наибольшего прироста функционала. Таким образом, параметр t^k может служить для регулирования сходимости алгоритма и выбирается с тем, чтобы обеспечивалось уменьшение функционала на k-ой итерации.

9.5. Особые (вырожденные) и скользящие режимы управления

Для поиска оптимального управления на основе принципа максимума (см. (9.7)) необходимо в каждый момент времени решать задачу нелинейного программирования вида

$$u^*(t) = \arg \max_{u \in G_u} H(t,\overline{x},u,\overline{\psi}).$$

В ситуациях, когда множество допустимых управлений G_u выпукло, замкнуто и ограничено, а функция Гамильтона вогнута, решение такой задачи нелинейного программирования при фиксированных t, x, ψ существует и единственно. В этом случае управление $u^*(t)$ удовлетворяет необходимым условиям существования оптимального управления и является экстремалью.

Вместе с тем при решении задачи максимизации функции Гамильтона по множеству допустимых управлений G_u возможны ситуации, когда

а) на некотором интервале времени $(t_1,t_2]$, $t_1,t_2 \in (t_o, t_f]$ функция Гамильтона не зависит от управления u;

б) на некотором интервале времени $(t_3,t_4]$, $t_3,t_4 \in (t_o,t_f]$ оптимальное управление не единственно, т.е. существует несколько управлений $u^*(t)$, доставляющих максимум функции Гамильтона.

Первая группа ситуаций приводит к так называемым особым [38,52] или вырожденным [5] режимам управления, вторая группа ситуаций - к скользящим [52] режимам управления. Поясним кратко особенности указанных режимов.

<u>Особый режим управления</u>.

Итак, пусть на некотором интервале времени $(t_1,t_2]$, $t_1,t_2 \in (t_o,t_f]$ гамильтониан $H(t,x,u,\psi)$, $t \in (t_1,t_2]$ не является функцией управления (не зависит от u), тогда

$$\frac{\partial H(t,x,u,\psi)}{\partial u} = 0. \tag{9.18}$$

В этом случае интервал времени $(t_1,t_2]$ называется участком особого (вырожденного [5]) режима управления; управление $u(t)$, $t \in (t_1,t_2]$ называется особым управлением.

Так как в соответствии (9.18) на участке особого управления частная производная функции Гамильтона по управлению, как функция времени, тождественно равна нулю, то функция $u(t)$ и функции $x(t),\psi(t)$ могут на этом участке принимать только такие значения, что

$$\frac{d^k}{dt^k}\left(\frac{\partial H}{\partial u}\right) = 0,\ k = 1,2,\dots\dots \tag{9.19}$$

Последнее выражение совместно с (9.18) позволяет выделить все участки особого режима.

Необходимое условие оптимальности особого управления при решении задачи на минимум функционала будет иметь вид [38]:

$$\frac{\partial}{\partial u}\left(\frac{d^2}{dt^2}\frac{\partial H}{\partial u}\right) \geq 0 \qquad (9.20)$$

Если же решается задача максимизации функционала, то знак неравенства в (9.20) меняется на противоположный.

В соответствии с (9.20) может выделяться некоторое множество особых управлений. Таким образом, существование особого режима приводит к тому, что задача оптимального управления при некоторых начальных условиях имеет множество оптимальных управлений.

Оптимальное в целом управление находится таким образом, что на обычных участках оно выбирается в соответствии с принципом максимума, а на особых участках выбирается одно из управлений, удовлетворяющих (9.20).

<u>Скользящий режим управления</u>.

В ситуациях, когда на некотором интервале времени $(t_3, t_4]$, $t_3, t_4 \in (t_o, t_f]$ оптимальное управление не единственно, т.е. существует несколько различных управлений $u^{*0}(t), u^{*1}(t), ..., u^{*k}(t)$, доставляющих максимум гамильтониану в момент t. Такие ситуации могут складываться либо, если функция Гамильтона многоэкстремальна, либо, если множество допустимых управлений G_u невыпукло. Такие ситуации приводят к, так называемым, скользящим режимам управления [38,52].

Проблема выбора оптимального управления при $t \in (t_3, t_4]$ заключается в том, что в соответствии с каждым, прикладываемым к системе, управлением $u^{*j}(t)$, $j=0,...,k$ изменяется и состояние динамической системы

$$\dot{x}(t) = \varphi(t, x, u^{*j}),$$

причем в каждый момент времени можно выбирать любое из $u^{*j}(t)$, $j=0,...,k$ управлений. При этом на интервале времени $t \in (t_3, t_4]$ можно строить различные последовательности управлений (а, следовательно, и фазовые траектории).

Разобьем промежуток $(t_3, t_4]$ на s частей точками τ_p, p=0, ..., s таких, что $t_3 = \tau_o < \tau_1 < .... < \tau_s = t_4$. Причем выполняется условие при $s \to \infty$ $\delta_s = \max\limits_p \left|\tau_{p+1} - \tau_p\right| \to 0,\ p = 0,...,s-1.$

Каждая точка $x_s(\tau_p)$ соединяется с точкой $x_s(\tau_{p+1})$ ломаной линией, составленной из отрезков

$$\varphi\,(t,\, x(\tau_p),\, u^{*j}(\tau_p))\cdot t\ ,\ j=0,...,k.$$

Построенная таким образом ломаная $x_s(t)$ является решением уравнений состояния. В целом же последовательность $\{x_s(t),\, u_s(t)\}$ удовлетворяет дифференциальным связям, доставляет минимум функционалу, и является минимизирующей последовательностью, которая при $s \rightarrow \infty$ стремится к оптимальному скользящему режиму.

Особенности реализации алгоритмов, позволяющих выявлять оптимальные скользящие режимы, рассмотрены в [52].

Контрольные вопросы

1. Какие различаются методы решения задач оптимального управления?
2. Какие методы решения задач оптимального управления строятся на основе использования принципа максимума Л.С.Понтрягина ?
3. В чем состоит суть краевой задачи, и в чем заключаются трудности ее решения ?
4. Каким образом обеспечивается решение краевой задачи на основе метода Ньютона? В чем его достоинства и недостатки ?
5. В чем заключаются особенности решения краевой задачи на основе метода Крылова-Черноусько? В чем достоинства и недостатки метода ?
6. В результате чего появляются вырожденные управления? Каким образом принимать решение в этом случае ?
7. В результате чего появляются скользящие режимы управления ?

10. АЛГОРИТМЫ ВЫБОРА ОПТИМАЛЬНОГО РЕШЕНИЯ В ФОРМЕ СИНТЕЗА УПРАВЛЕНИЯ

10.1. Постановка задачи синтеза управления

Рассмотренные в предыдущем разделе методы и алгоритмы позволяют найти оптимальное управление в форме некоторой функции от времени $u(t)$, такое управление называют программным управлением и оно позволяет перевести динамическую систему из заданного начального состояния в конечное состояние с некоторым наилучшим качеством, задаваемым функционалом. Вместе с тем в ряде приложений целесообразно находить управление, не привязанное к конкретному начальному состоянию, но являющееся функцией от состояния $u(t,x(t))$ или функцией от выхода $u(t,y(t))$, т.е. в форме синтеза управления или в форме управления с обратной связью.

Разработка вычислительных методов синтеза представляется весьма перспективным направлением развития управления современными автоматизированными системами. Реализация управления в форме синтеза позволяет более успешно решать целевые задачи в условиях неопределенности обстановки, повышать устойчивость управления. Это связано, прежде всего, с тем, что реальные условия функционирования системы могут отличаться от тех, которые были учтены в математической модели, использованной при поиске оптимального программного управления. Тогда реальное состояние системы в момент t может быть отлично состояния, рассчитанного на модели, а, следовательно, и управление, найденное как функция времени, не отвечает сложившимся (или изменившимся) условиям. В отличие от программного управления оптимальное управление $u^*(t,x(t))$, рассчитанное в форме синтеза, в каждый момент времени t зависит только от того, каково конкретное состояние динамической системы. Поэтому для любого допустимого состояния x(t), которое приняла динамическая система в результате того или иного внешнего воздействия, известно управление, позволяющее из данного состояния x(t) достигать заданного конечного состояния наилучшим образом. Интуитивно это и понятно, действительно, оптимальное управление в момент t зависит от состояния системы, а не от предыстории процесса (начального состояния и того, каким образом происходило движение системы до момента t).

Математически задачу синтеза можно сформулировать следующим образом. Необходимо найти функцию управления $u(t,x(t))$, $t \in T = (t_o,t_f]$, доставляющую минимум функционалу

$$J(t,x,u) \rightarrow \min , \tag{10.1}$$

и удовлетворяющую ограничениям вдоль траектории

$$u(t,x(t)) \in G_u \subseteq R^m , \tag{10.2}$$

и которая переводит динамическую систему

$$\dot{x}(t) = \varphi(t, x, u(t, x(t))) \tag{10.3}$$

в конечное состояние

$$x(t_f) \in E_f \tag{10.4}$$

из некоторого начального состояния.

Сформулированная задача похожа на задачу поиска оптимального программного управления (8.25)-(8.28). Существенным отличием является то, что в задаче синтеза значения $x(t_o)$ и t_o являются произвольными. Следовательно, необходимо построить управление, переводящее систему (10.3) в конечное состояние $x(t_f) \in E_f$ из любого начального состояния $x(t_o)$ для любого момента времени t_o.

В общем случае задача синтеза (т.е. вопрос о существовании управления в форме синтеза и способы его нахождения) не решена.

10.2. Синтез управления в линейных динамических системах

Наиболее простыми задачами синтеза оптимального управления, поставленными в 10.1, являются задачи синтеза линейных (8.5) систем с линейным или квадратичным функционалом. Для таких систем выполняются условия теоремы существования и единственности, т.е. для каждого $x_o = x(t_o)$ существует (и только одно) управление $u(t,x_o)$, переводящее систему из состояния x_o в заданное состояние $x_f = x(t_f)$.

Для любого момента времени $\tau \in (t_o, t_f]$ управление $u(t, x_o)$, рассматриваемое на интервале $(\tau, t_f]$ должно быть оптимальным, тогда можно обозначить

$$u(\tau, x_o) = v(x(\tau)).$$

В этом случае (8.4) можно переписать

$$\dot{x} = A(t)\, x + B(t)\, v(x).$$

Решение данного уравнения с произвольно заданным x_o позволяет найти функцию $v(x)$, которая синтезирует оптимальное управление, переводящее систему из x_o в x_f. В нахождении такой функции $v(x)$ и заключается задача синтеза для линейных систем.

Рассмотрим особенности синтеза оптимального управления для линейной системы в задаче Лагранжа с квадратичным функционалом

$$J(x, u) = \frac{1}{2}\int_{t_o}^{t_f} \left(x^{T}(\tau) Q(\tau) x(\tau) + u^{T} R(\tau) u(\tau) \right) dt \to \min,$$

здесь $Q(t)$ - неотрицательно-определенная симметрическая мат рица;

$R(t)$ - положительно определенная симметрическая матрица для $\forall t$.

Тогда $x^{T}(t)Q(t)x(t) \geq 0$, $u^{T}(t)R(t)u(t) > 0$ при $\forall t \in (t_o, t_f]$.

В соответствии с принципом максимума Л.С. Понтрягина оптимальное управление в любой момент времени находится, исходя из максимизации функции Гамильтона (8.43)

$$H = \frac{1}{2}\psi_o\, x^{T}Q(t)x + \frac{1}{2}\,\psi_o\, u^{T}R(t)u + \psi^{T}A(t)x + \psi^{T}\, B(t)u,$$

или, транспонируя обе части равенства,

$$H = \frac{1}{2}\psi_o\, x^{T}Q(t)\, x + \frac{1}{2}\,\psi_o\, u^{T}R(t)u + x^{T}A(t)\psi + u^{T}B(t)\psi$$

Обозначим $p_i(t) = \dfrac{\psi_i(t)}{\psi_0}, i = 1, ..., n.$

Тогда в соответствии с (8.38)

$$\dot{\psi}_i = \psi_o\, \dot{p}_i = -\frac{dH}{dx_i}, i = 1, ..., n.$$

Откуда

$$\dot{p} = -Q(t)\,x - A^{T}(t)p. \tag{10.5}$$

Оптимальное управление должно доставлять максимум функции Гамильтона. Необходимым условием этого является равенство нулю частной производной гамильтониана по управлению, откуда

$$R(t)\,u + B^{T}(t)\,p = 0.$$

Тогда

$$u = -R^{-1}(t)\,B^{T}(t)\,p.$$

Подставляя последнее выражение для управления в дифференциальные уравнения (8.4), описывающие изменение состояния, получим

$$\dot{x} = A(t)x - B(t)R^{-1}(t)B^{T}(t)\,p\,. \tag{10.6}$$

Будем искать неизвестную функцию p(t) в виде

$$p(t) = K(t)\,x(t), \tag{10.7}$$

где $K(t)$ - неизвестная матрица размерности $n \times n$.
Тогда (10.6) можно переписать в виде

$$\dot{x} = A(t)x - B(t)R^{-1}(t)B^{T}K(t)x\,.$$

Подставляя (10.7) в (10.5), с учетом последнего выражения можно получить

$$\dot{K}(t)x + K(t)\dot{x} = \dot{K}(t)x + K(t)A(t)x - K(t)B(t)R^{-1}(t)B^{T}(t)K(t)x =$$

$$= -Q(t)x - A^{T}(t)K(t)x.$$

Отсюда неизвестная в (10.7) матрица K(t) должна удовлетворять матричному дифференциальному уравнению вида

$$\dot{K}(t) = -K(t)A(t) - A^{T}(t)K(t) + K(t)B(t)R^{-1}(t)B^{T}(t)K(t) - Q(t).$$

Данное нелинейное матричное дифференциальное уравнение, известное как уравнение Риккати, позволяет найти матрицу $K(t)$ для любого момента времени $t \in (t_o, t_f]$ при начальных условиях, вытекающих из условий трансверсальности. Тогда оптимальное управление $u^*(t)$ для состояния $x(t)$ определяется как

$$u^*(t) = -R^{-1}(t)\,B^{T}(t)\,K(t)\,x(t).$$

Для стационарной линейной динамической системы, описываемой уравнениями (8.5), дифференциальное уравнение Риккати превращается в нелинейное матричное уравнение вида

$$KA + A^{т}K - KBR^{-1}B^{т}K + Q = 0.$$

Решение уравнений Риккати базируется, как правило, на использовании итеративных численных методов.

10.3. Синтез управления в рекуррентных системах. Принцип оптимальности. Беллмана.

На основе поставленной в 8.7 задачи оптимального программного управления рекуррентными системами (8.56)-(8.59) может быть сформулирована задача синтеза в следующем виде

$$J(x,u) = \sum_{k=0}^{N-1} F\big((x(k), u(k, x(k))\big) \to \min , \tag{10.8}$$

$$x(k+1) = f(x(k),u(k),x(k)),\ k = 0,...,N-1, \tag{10.9}$$

$$u(k,x(k)) \in G_u \subseteq R^m, \qquad k = 0,...,N-1, \tag{10.10}$$

$$x(N) = x_f. \tag{10.11}$$

Для поиска оптимального управления u*(k,x(k)) в рекуррентных системах часто используются, алгоритмы основанные на принципе оптимальности Р.Беллмана. Принцип оптимальности, сформулированный для непрерывных систем, представляет собой развитие принципа причинности для динамических систем (см.8.1). Приведем формулировку принципа оптимальности и некоторые следствия, вытекающие из него и поясняющие данный принцип.

Принцип оптимальности.

Оптимальное управление в любой момент времени t, обладает тем свойством, что каково бы ни было начальное состояние динамической системы и управления в моменты времени, предшествующие t, последующие управления должны быть оптимальны относительно состояния динамической системы в момент времени t.

<u>Следствие 1</u>.

Оптимальное управление в любой момент времени t не зависит от предыстории системы и определяется только состоянием системы в момент t и целью управления.

<u>Следствие 2</u>.

В любой момент времени t ($t0 \le t < t_f$) участок оптимальной траектории от x(t) до (t_f) сам по себе является оптимальной траекторией.

<u>Следствие 3</u>.

Если в некоторый момент времени управление не оптимально, то последствия этого отклонения от оптимального управления нельзя исправить в будущем.

Поясним принцип оптимальности в задаче (10.8)-(10.11). Пусть имеется (см. рис.10.1) оптимальная траектория, переводящая систему из состояния x(0) в состояние x(N) и доставляющая минимум функционалу J(x,u). Пусть точка x(s) разбивает рассматриваемую траекторию на два участка: участок 1 - x(k), k=0,...,s-1; участок 2 - x(k), k=s,...,N. Участок 2 может рассматриваться и как самостоятельная траектория. Эта траектория будет оптимальной, если она доставляет минимум функционалу.

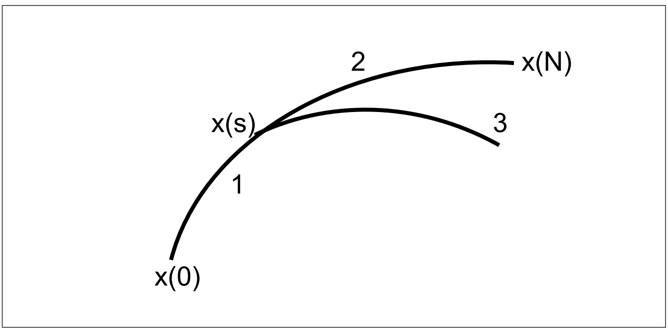


Рис.10.1.

В соответствии со следствием 2 принципа оптимальности - участок 2 оптимальной траектории 1-2 сам по себе является оптимальной траекторией.

Предположим противное, пусть существует другая траектория 3, отличная от траектории 2 и доставляющая функционалу

J(x,u) значение, меньшее, чем на траектории 2. Но тогда на интервале от 0 до N оптимальной будет не траектория 1-2, а траектория 1-3. Последнее противоречит исходному предположению об оптимальности траектории 1-2. Следовательно, участок 2 должен быть оптимальной траекторией, как это и следует из принципа оптимальности.

На основе приведенного принципа оптимальности Р. Беллманом был разработан алгоритм динамического программирования, показавший достаточно высокую эффективность в задачах оптимизации рекуррентных систем и позволяющий получать оптимальное управление в форме синтеза.

10.4. Алгоритм динамического программирования

Пусть состояние x(k) на любом шаге k, k=1,...,N-1 может принимать значение из конечного множества состояний (на шаге N в соответствии с (10.11) это множество состоит из одного элемента $x(N)=x_f$); обозначим множество различных состояний на шаге k, как X_k. Отметим, что число таких состояний может быть достаточно велико, так, например, если x(k) - вектор, состоящий из четырех компонент, и каждая компонента может принимать всего 3 значения, то при любом k значение $|X_k| = 3^4 = 81$.

Суть алгоритма динамического программирования состоит в том, что, построение оптимального управления, как функции состояния, производится из конца траектории к началу. Начиная с предпоследнего (k = N -1) шага, для каждого допустимого состояния x(k) (число которых равно $|X_k|$) рассчитывается оптимальное управление u*(k,x(k)), которое согласно принципу оптимальности соответствует оптимальной траектории x*(k).

Итак, пусть на шаге (N-1) система находится в некотором состоянии x(N-1), тогда оптимальное управление u*(N-1,x(N-1)) должно в соответствии с (10.9)-(10.11) переводить систему в состояние $x(N) = x_f$ и доставлять минимум функционалу (10.8), т.е. оно находится в результате решения задачи

$$F(x(N-1),u(N-1,x(N-1))) \to \min_{u(N-1,x(N-1))\in G_u}$$

Обозначим минимальное значение функционала через $S_{N-1}(x(N-1))$, тогда последнее выражение можно переписать в виде

$$S_{N-1}\left(x(N-1)\right) = \min_{u(N-1,x(N-1))\in G_u} F\left(x(N-1), u(N-1, x(N-1))\right), \quad (10.11)$$

При фиксированном значении x(N-1) минимизируемая функция в (10.11) зависит только от u(N-1,x(N-1)), откуда и находится оптимальное управление u*(N-1,x(N-1)).

Рассмотрим теперь предыдущий интервал, шаг (N-2). Пусть система находится в состоянии x(N-2), тогда задачу поиска оптимального управления в точке (N-2), с учетом (10.11), (10.8) можно представить в виде

$$S_{N-2}\left(x(N-2)\right) =$$
$$= \min_{\substack{u(N-2,x(N-2))\in G_u \\ u(N-1,x(N-1))\in G_u}} \left(F\left(x(N-2), u(N-2, x(N-2))\right) + F\left(x(N-1), u(N-1, x(N-1))\right)\right)$$

Поскольку первое слагаемое под знаком минимума не зависит от состояния и управления на шаге (N-1), то последнее выражение можно переписать в виде

$$S_{N-2}(x(N-2)) = \min_{u(N-2,x(N-2))\in G_u} (F(x(N-2),u(N-2,x(N-2))) +$$
$$+ \min_{u(N-1,x(N-1))\in G_u} F(x(N-1),u(N-1,x(N-1)))).$$

Второе слагаемое в соответствии с (10.11) равно S_{N-1}(x(N-1)), тогда

$$S_{N-2}(x(N-2)) = \min_{u(N-2,x(N-2))\in G_u} (F(x(N-2),u(N-2,x(N-2))) + S_{N-1}(x(N-1)))$$

Тогда в целом для произвольного шага (N-h) значение функции S_{N-h}(x(N-h))

$$S_{N-h}(x(N-h)) =$$
$$= \min_{u(N-h,x(N-h))\in G_u} (F(x(N-h),u(N-h,x(N-h))) + S_{N-h+1}(x(N-h+1))) \quad (10.12)$$

или с учетом (10.9)

$$S_{N-h}(x(N-h)) =$$
$$= \min_{u(N-h,x(N-h))\in G_u} (F(x(N-h),u(N-h,x(N-h))) + S_{N-h+1}(f(x(N-h),u(N-h)))) \quad (10.13)$$

В выражении (10.13) поиск минимума функции S_{N-h}(x(N-h)) выполняется по одной переменной u(N-h) = u(N-h,x(N-h)). Полученное при этом значение u(N-h) и будет искомым оптимальным управлением, найденным, как функция состояния u*(N-h,x(N-h)).

Функция $S_{N-h}(x(N-h))$ часто называется функцией Беллмана, а уравнение (10.13) рекуррентным уравнением Беллмана.

Итак, начиная с шага N-1 (h=1) находится оптимальное управление $u^*(N-h,x(N-h))$, которое вычисляется, исходя из минимизации функции Беллмана $S_{N-h}(x(N-h))$; процесс вычислений заканчиваются, когда h = N.

Замечание.

В ситуациях, когда задано исходное состояние x(0), можно вычислить оптимальное программное управление, как функцию шага k=0,..., N-1 (k=N-h). Действительно, поскольку известно оптимальное управление в форме синтеза u*(N-h,x(N-h)), h=1,..,N, то при h=N (k=0) и известном состоянии x(0) можно вычислить оптимальное управления u*(0). Тогда из выражения (10.9) можно определить состояние системы x(1) (при k=1 (h=N-1)). Для состояния x(1) выбирается оптимальное программное управление u*(1), уже рассчитанное ранее. Теперь из (10.9) можно найти состояние x(2) и так далее... Таким образом, формируется оптимальное программное управление u*(k), k=0,..., N-1.

Алгоритм.

0). Исходное состояние h = 1. Конечное состояние системы задано $x(N)=x_f$. Система находится в состоянии x(N-h) = x(N-1);

$S_{N-h+1}(x(N-h+1))) = S_N(x(N)) = 0$; $X_{N-h+1} = \{x(N)\}$

1). Рассчитывается множество состояний X_{N-h}, число которых равно $|X_{N-h}|$.

2). Для каждой пары состояний $x(N-h)\in X_{N-h}$ и $x(N-h+1)\in X_{N-h+1}$ находятся различные управления $u(N-h,x(N-h))\in G_u$, позволяющие перевести систему из состояния x(N-h) в состояние x(N-h+1). Для каждого такого управления рассчитывается значение функции

$$Z(x(N-h),u(N-h,x(N-h)),x(N-h+1)) =$$
$$= (F(x(N-h),u(N-h,x(N-h))) + S_{N-h+1}(x(N-h+1))) .$$

3). Производится выбор оптимального управления из условия

$$u^*(N-h,x(N-h)) = \arg \min_{u(N-h,x(N-h))} Z(x(N-h),u(N-h,x(N-h)),x(N-h+1))$$

Вычисляется значение функции Беллмана

$$S_{N-h}(x(N-h)) = \min_{u(N-h,x(N-h))} Z(x(N-h),u(N-h,x(N-h)),x(N-h+1))$$

В результате выполнения данного шага для каждого допустимого в момент (N-h) состояния x(N-h) вычисляется оптимальное управление u*(N-h,x(N-h)), состояние x(N-h+1), в которое перейдет система под воздействием данного управления, и значение функции Беллмана S_{N-h}(x(N-h)).

4). h = h + 1. Если (N - h) < 0 - процесс счета заканчивается; в противном случае производится переход на шаг 1.

Рассмотренный алгоритм динамического программирования может быть распространен на задачи синтеза оптимального управления для непрерывных динамических систем, описываемых дифференциальными уравнениями (доказательство этого было проведено В.Г.Болтянским). При этом рекуррентное уравнение (10.13) преобразуется в нелинейное дифференциальное уравнение в частных производных. При выполнении определенных требований непрерывности частных производных по всем своим аргументам (что, кстати, довольно часто не выполняется, даже для простых задач) может быть найдено оптимальное управление в форме синтеза.

Контрольные вопросы

1. В каком виде ищется оптимальное решение в задачах синтеза оптимального управления ?
2. В чем заключаются достоинства практического использования управления в виде синтеза ?
3. В чем состоят трудности нахождения оптимального решения в форме синтеза управления ?
4. Каким образом найти оптимальное управление в форме синтеза, если изменение состояния динамической системы описывается линейными дифференциальными уравнениями, а функционал является квадратичным ?
5. В чем состоит суть принципа оптимальности ?
6. В чем заключаются основные особенности поиска оптимального решения с использованием алгоритма динамического программирования ?

11. ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

11.1. Оптимальное управление сближением КА

В 8.4.1. рассматривалась постановка задачи оптимального управления сближением КА. Первым вопросом, который возникает при формализации задачи принятия решения с использованием динамических моделей, является определение понятия состояния динамической системы. В зависимости от того, какой состав характеристик выбран в качестве состояния, определяется вид дифференциальных уравнений, описывающих изменение данного состояния во времени. Так в 8.4.1 состояние описывалось вектором x(t), состоящим из двух компонент $x(t) = \| x_1(t)\ x_2(t) \|^{т} = \| r(t)\ v(t) \|^{т}$, где r(t) - скалярная функция времени, описывающая изменение относительной дальности между КА; v(t) - функция, описывающая изменение относительной скорости. В этом случае дифференциальные уравнения, описывающие изменение состояния, имеют вид

$$\dot{x}_1 = x_2 \ ,$$

$$\dot{x}_2 = u \ .$$

Отметим, что, если бы в качестве состояния были выбраны параметры положения КА и его скорости в абсолютной геоцентрической системе координат (ортогональная система, центр которой соответствует центру Земли), то движение того же самого объекта описывалось бы шестью дифференциальными уравнениями.

Величина прикладываемого импульса u(t) в любой момент времени ограничивается техническими характеристиками двигательной установки

$$u(t) \in [-U_o, +U_o] \ .$$

Начальное состояние системы $x(t_o)$ соответствует некоторому взаимному положению КА в пространстве

$$x(t_o) = \| r(t_o)\ v(t_o) \|^{T} = \| r_o\ v_o \|^{T}.$$

Конечное состояние соответствует ситуации, когда КА сблизились, тогда

$$x(t_f) = \| 0\ 0 \|^{T}.$$

В этих условиях необходимо найти управление, доставляющее оптимальное значение некоторому функционалу.

11.1.1. Управление КА, оптимальное по быстродействию

Оптимальное по быстродействию управление соответствует сближению за минимальное время. Будем без ограничения общности считать, что $t_o = 0$, $t_f = T$, тогда функционал, описывающий качество управления имеет вид

1.

$$J = \int_0^T dt \to \min. \tag{11.1}$$

2. Дифференциальные связи

$$\dot{x}_1 = x_2\ , \tag{11.2}$$

$$\dot{x}_2 = u\ . \tag{11.3}$$

3. Ограничения вдоль траектории

$$u(t) \in G_u = [-U_o, +U_o],\ \ t \in (0,T]. \tag{11.4}$$

4. Краевые условия

$$x(0) = \| r_o\ v_o \|^{T}, x(T) = \|\ 0\ 0\ \|^{T}. \tag{11.5}$$

Решение поставленной задачи на основе принципа максимума Л.С. Понтрягина (см. 8.6.2) основывается на том, что оптимальное управление обладает тем свойством, что функция Гамильтона в любой момент времени достигает максимума по управлению. Функция Гамильтона определяется в соответствии (8.36) как

$$H(t, \overline{x}, u, \overline{\psi}) = (\ \overline{\varphi},\ \overline{\psi}\ ) = \ \overline{\psi}^{T}\ \overline{\varphi}\ = \sum_{i=0}^{2} \psi_i\, \varphi_i$$

Для рассматриваемой задачи (11.1)-(11.5)

$$\overline{x}(t) = (x_o(t), x_1(t), x_2(t)),$$

где $x_o(t)$ характеризует значение функционала; $x_o(t) = J$.

Тогда дифференциальные уравнения (11.2),(11.3) необходимо дополнить уравнением вида

$$\dot{x}_o = 1. \qquad (11.2')$$

В этом случае вектор-функция правых частей дифференциальных уравнений (11.2'), (11.2),(11.3) имеет вид

$$\overline{\varphi} = \begin{Vmatrix} 1 \\ x_2 \\ u \end{Vmatrix}. \qquad (11.6)$$

Дифференциальные уравнения являются линейными, следовательно, их можно представить в виде

$$\frac{dx}{dt} = A\overline{x} + Bu + C = \begin{Vmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{Vmatrix} \overline{x} + \begin{Vmatrix} 0 \\ 0 \\ 1 \end{Vmatrix} u + \begin{Vmatrix} 1 \\ 0 \\ 0 \end{Vmatrix},$$

тогда сопряженная система дифференциальных уравнений будет иметь вид

$$\frac{d\psi}{dt} = -A^{т}\overline{\psi} = -\begin{Vmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{Vmatrix} \overline{\psi},$$

или

$$\dot{\psi}_o = 0, \quad \text{тогда} \quad \psi_o = \text{const} = \psi_{oo},$$
$$\dot{\psi}_1 = 0, \quad \text{откуда} \quad \psi_1 = \text{const} = \psi_{1o},$$
$$\dot{\psi}_2 = -\psi_1, \text{тогда} \quad \psi_2 = \psi_{2o} - \psi_{1o}t,$$

или $$\overline{\psi} = \begin{Vmatrix} \psi_{oo} \\ \psi_{1o} \\ \psi_{2o} - \psi_{1o}t \end{Vmatrix}$$

В этом случае функция Гамильтона имеет вид

$$H(t, \overline{x}, u, \overline{\psi}) = \overline{\psi}^{т}\overline{\varphi} = \psi_{oo} + \psi_{1o}x_2 + (\psi_{2o} - \psi_{1o}t)u$$

В соответствии с (8.45) оптимальное управление в любой момент времени t ищется, исходя из максимизации функции Гамильтона

$$H(t,\bar{x},u^*,\bar{\psi}) = \max_{u\in G_u} H(t,\bar{x},u,\bar{\psi}) =$$

$$= \max_{u\in G_u} (\psi_{oo} + \psi_{1o}x_2 + (\psi_{2o} - \psi_{1o}t)u) = \max_{u\in G_u} (\psi_{2o} - \psi_{1o}t)u$$

откуда оптимальное управление

$$u^*(t) = U_o \operatorname{sign}(\psi_{2o} - \psi_{1o} t). \qquad (11.6)$$

Для того, что бы определить вид функции оптимального управления $u^*(t)$ необходимо выяснить значения констант ψ_{1o}, ψ_{2o} в выражении (11.6).

Пусть в краевых условиях (11.5) $r_o > 0$, $v_o > 0$. Тогда при $t = 0$ из (11.6) следует, что знак $u^*(0)$ определяется знаком ψ_{2o}. Но, если $v_o > 0$, то объекты расходятся, следовательно, при $t=0$ необходимо давать тормозной импульс, т.е. управление $u^*(0)$ должно быть меньше 0, а значит константа $\psi_{20} <0$. Далее, из условий трансверсальности (8.46) следует, что

$$\psi_o(T) = \psi_{oo} \le 0 \ ; \ H(T,x,u^*,\psi) = 0.$$

Или

$$H(T,x,u^*,\psi) = \psi_{oo} + (\psi_{2o} - \psi_{1o} T) U_o = 0,$$

откуда

$$\psi_{1o} = \frac{\psi_{oo} + \psi_{2o} U_o}{T U_o} < 0 .$$

Тогда функции $\psi_2(t)$ и $u^*(t)$ имеют вид

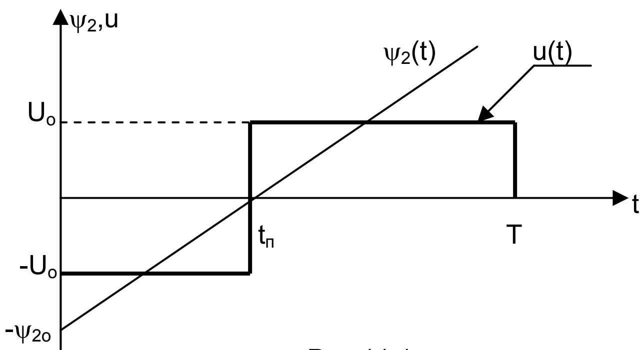


Рис.11.1.

Чтобы полностью определить оптимальное управление необходимо найти время переключения двигательной установки $t_п$.

Решим с этой целью уравнения состояния (11.2),(11.3) при начальных условиях (11.5).

$$\dot{x}_1 = x_2, \quad x_1(0) = x_{1o} = r_o,$$

$$\dot{x}_2 = u. \quad x_2(0) = x_{2o} = v_o.$$

Из второго уравнения получаем: $x_2(t) = x_{2o} + u\,t$,

тогда из первого уравнения имеем: $x_1(t) = x_{1o} + x_{2o}t + \dfrac{ut^2}{2}$.

Подставим в полученные уравнения начальные условия, тогда

$$x_1(t) = r_o + v_o t + \frac{ut^2}{2}. \tag{11.7}$$

$$x_2(t) = v_o + u\,t. \tag{11.8}$$

Выразим из второго уравнения время t через $x_2(t)$, и подставим его в первое уравнение, получим

$$x_1(t) = r_o + \frac{v_o x_2(t)}{u} - \frac{v_o^2}{u} + \frac{x_2^2(t) - 2v_o x_2(t) + v_o^2}{2u}$$

или

$$x_1(t) = \frac{x_2^2(t)}{2u} + r_o - \frac{v_o^2}{2u}\ . \tag{11.9}$$

Анализируя (11.9), можно заключить, что в фазовом пространстве (см. 8.2.1.) траектории движения системы при различных начальных условиях и управлениях имеют вид парабол

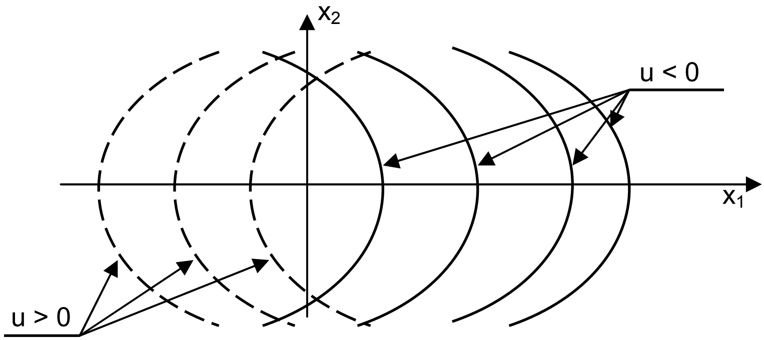


Рис.11.2.

Для рассматриваемого случая фазовая траектория имеет вид

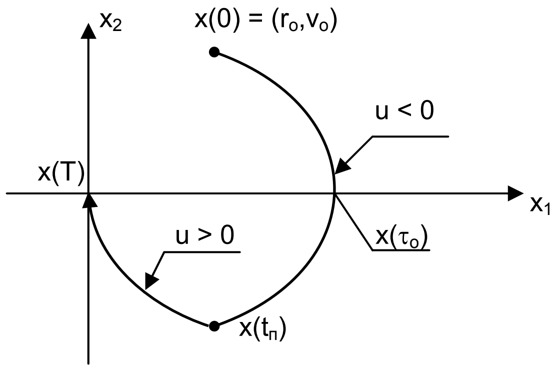


Рис.11.3.

Для вычисления момента времени переключения двигательной установки $t_п$ рассмотрим движение системы по участкам:

- участок 1: $t \in (0\ ,\ \tau_o]$;
- участок 2: $t \in (\tau_o,\ t_п]$;
- участок 3: $t \in (t_п\ ,\ T\ ]$.

При этом будем иметь в виду то, что $t_п = \tau_o + \tau_п$; $T = t_п + \tau_м = \tau_o + \tau_п + \tau_м$, где τ_o – длительность участка 1, $\tau_п$ - длительность участка 2, $\tau_м$ – длительность участка 3.

<u>Участок 1</u>: $(r_o, v_o) \xrightarrow{\tau_0} (x_1^o, 0)$.

Так как $x_2(\tau_o) = 0$, то из уравнения (11.7) с учетом того, что $u = -U_o$ получаем

$$x_1(\tau_o) = x_1^o = r_o + \frac{v_o^2}{2U_o}. \tag{11.10}$$

Из (11.8) получаем, что

$$\tau_o = \frac{v_o}{U_o}. \tag{11.11}$$

Участок 2: $(x_1(\tau_o), 0) \xrightarrow{\tau_п} (x_1(t_п),\ x_2(t_п))$.

В соответствии с (11.7) и с учетом того, что на этом участке $u = -U_o$

$$x_1(t_п) = x_1(\tau_o) + 0\,\tau_п - \frac{U_o r_п^2}{2}$$

или, подставляя (11.10) в последнее выражение, получим

$$x_1(t_п) = r_o + \frac{v_o^2}{2U_o} - \frac{U_o \tau_п^2}{2}\ . \qquad (11.12)$$

В соответствии с (11.8) и с учетом того, что $u = -U_o$

$$x_2(t_п) = 0 - U_o\tau_п$$

Участок 3: $(x_1(t_п),\ x_2(t_п)) \xrightarrow{\tau_м} (0, 0)$).

Движение системы на этом участке, в отличие от участков 1, 2, осуществляется при $u = U_o$. Из рис. 11.2, 11.3 и выражения (11.9) можно заключить, что траектория движения проходит по параболе, аналогичной участку 2, но ориентированной в противоположную сторону. Тогда, из того, что $x_2(T) = 0$, следует, что $\tau_м = \tau_п$. Будем иметь в виду данный факт. В этом случае из выражения (11.7) и того, что $u = +U_o$, получаем

$$0 = x_1(t_п) + x_2(t_п) + \frac{U_o\tau_п^2}{2}\ , \qquad (11.13)$$

из (11.8) получаем

$$0 = x_2(t_п) + U_o\,\tau_п. \qquad (11.14)$$

Подставляем в (11.13) выражение для $x_1(t_п)$ из (11.12) и выражение для $x_2(t_п)$ из (11.14). Тогда

$$0 = r_o + \frac{v_o^2}{2U_o} - \frac{U_o\tau_п^2}{2} - U_o\tau_п^2 + \frac{U_o\tau_п^2}{2}$$

откуда

$$\tau_п = \sqrt{\frac{r_o}{U_o} + \frac{v_o^2}{2U_o^2}}\ . \qquad (11.15)$$

Тогда с учетом (11.11) и (11.15)

$$t_п = \tau_o + \tau_п = \frac{v_o}{U_o} + \sqrt{\frac{r_o}{U_o} + \frac{v_o^2}{2U_o^2}}\ . \qquad (11.16)$$

В этом случае время сближения КА определяется как

$$T = t_п + \tau_м = \tau_о + 2\tau_п = \frac{v_о}{U_о} + 2\sqrt{\frac{r_о}{U_о} + \frac{v_о^2}{2U_о^2}}$$

Таким образом, оптимальное управление в рассматриваемой задаче определяется видом функции u(t), представленной на рис.11.1, причем $t_п$ определяется выражением (11.16).

Пример.

С целью проведения оперативной инспекции технического состояния КА необходимо произвести сближение с ним за минимальное время. При этом расстояние между КА равно 60 км, а относительная скорость 40 м/сек. На КА-инспекторе установлена двигательная установка мощностью 2 м/сек^2. Необходимо найти оптимальное управление.

Вид оптимального управления в данной задаче соответствует управлению, представленному на рис. 11.1. Так как $r_о$ = 60 км = 60 000 м > 0 и $v_о$ = 40 м/сек > 0, то для расчета момента времени переключения двигательной установки с отрицательного импульса на выдачу положительного управления можно воспользоваться формулой (11.16).

$$t_п = \frac{40}{2} + \sqrt{\frac{60000}{2} + \frac{1600}{2 \cdot 4}} = 20 + 173.8 = 193.8$$

Таким образом, $t_п$ = 193.8 сек, при этом минимальное время сближения равно T = 367.6 сек.

11.1.2. Оптимальное по расходу топлива управление КА

Расход топлива двигательной установки зависит от величины прикладываемого управляющего импульса и длительности работы двигательной установки. Тогда функционал в рассматриваемой задаче (11.1)-(11.5) имеет вид

$$J = \int_0^T |u(t)|\; dt \;\to \min. \qquad (11.17)$$

Время T задано, причем оно не должно быть меньше минимального времени, полученного в задаче о максимальном быстродействии (см. 11.1.1).

Поскольку в данной задаче рассматривается та же динамическая система, что и предыдущем пункте, то расширение исходной системы и построение сопряженной динамической системы проводится аналогично тому, как это делалось в 11.1.1.

Из условий трансверсальности (8.46) следует, что $\psi_o(T)\le 0$. Пусть $\psi_o(T) = -1$. Тогда функция Гамильтона (см.(8.43) в рассматриваемой задаче будет иметь вид

$$H=\psi_o F(t,x,u)+\sum_{i=1}^{n}\psi_i\varphi_i\,(t,x,u)=-\left|u\right| + \psi_{1o}x_2+(\psi_{2o}-\psi_{1o}t)u\,.$$

Оптимальное управление u*(t), как это следует из принципа максимума Л.С. Понтрягина, обладает тем свойством, что оно доставляет максимум функции Гамильтона в каждый момент времени

$$u^*(t)=\arg\max_{u\in G_u}(-\left|u\right|+(\psi_2(t)u))=\max_{u\in G_u}\left(-\left|u\right|+(\psi_{2o}-\psi_{1o}t)u)\right)$$

Откуда

$$u^* = \begin{cases} U_o\,\mathrm{sign}\,\psi_2(t), & \text{при } |\psi_2(t)|>1 \\ 0, & \text{при } |\psi_2(t)|\le 1 \end{cases}$$

Тогда оптимальное управление при $r_o>0$, $v_o>0$ имеет вид

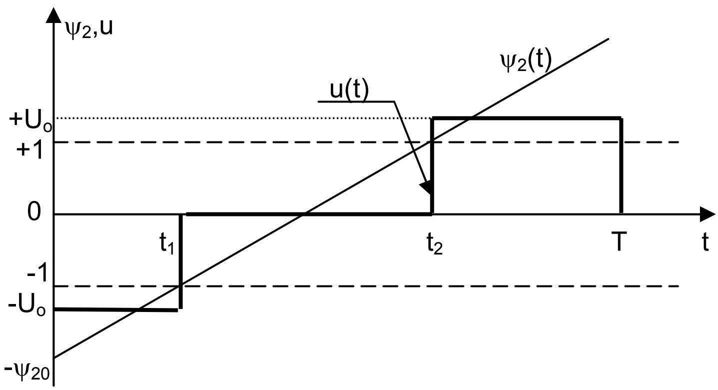


Рис. 11.4.

Оптимальное управление в данной задаче характерно тем, что на участке $t\in(0,t_1]$ двигательная установка работает на торможение (с целью погасить начальную скорость v_o); на участке

$t \in (t_1, t_2]$ двигательная установка выключена, и объекты сближаются с некоторой постоянной скоростью $x_2(t_1)$; на участке $t \in (t_2, T]$ двигатель включается с целью гашения скорости сближения.

Фазовая траектория в этом случае имеет вид

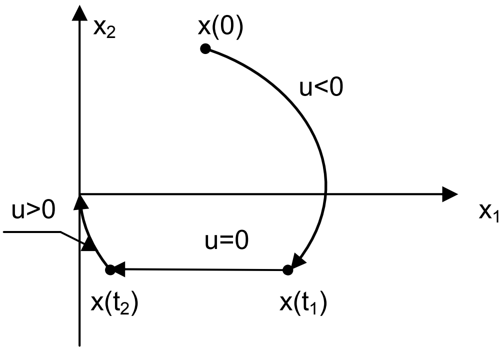


Рис. 11.5.

Введем следующие обозначения:

$\tau_1 = t_1$; $\tau_2 = t_2 - t_1$; $\tau_3 = T - t_2$; $x^1 = x(t_1)$; $x^2 = x(t_2)$.

Откуда заданное время сближения определяется как

$T = \tau_1 + \tau_2 + \tau_3$.

Рассмотрим движение системы по участкам, при этом будем использовать уравнения движения (11.7), (11.8)

$$x_1(t) = x_{1o} + x_{2o}t + \frac{u t^2}{2}.$$

$$x_2(t) = x_{2o} + u t.$$

Участок 1: $(r_o, v_o) \xrightarrow{\tau_1} (x_1^1, x_2^1)$, $u = -U_o$.

$$x_1^1 = r_o + v_o\tau_1 - \frac{U_o\tau_1^2}{2}, \qquad (11.18)$$

$$x^1_2 = v_o - U_o\tau_1. \qquad (11.19)$$

Участок 2: $(x_1^1, x_2^1) \xrightarrow{\tau_2} (x_1^2, x_2^2)$, $u = 0$.

$$x^2_1 = x^1_1 + x^1_2\tau_2, \qquad (11.20)$$

$$x^2_2 = x^1_2. \qquad (11.21)$$

Участок 3: $(x_1^2, x_2^2) \xrightarrow{\tau_3} (0, 0)$, $u = +U_0$.

$$0 = x_1^2 + x_2^2 \tau_3 + \frac{U_o \tau_3^2}{2}. \qquad (11.22)$$

$$0 = x^2{}_2 - U_o\tau_3. \qquad (11.23)$$

Подставим (11.19) в (11.21) и результат в (11.23), тогда получим

$$\tau_3 = \tau_1 - \frac{v_o}{U_o} \qquad (11.24)$$

Так как $T = \tau_1 + \tau_2 + \tau_3$, то из (11.24) получаем

$$\tau_2 = T - 2\tau_1 + \frac{v_o}{U_o}. \qquad (11.25)$$

Подставим (11.18), (11.19) в (11.20), тогда

$$x_1^2 = r_o + v_o\tau_1 - \frac{U_o\tau_1^2}{2} + (v_o - U_o\tau_1)\tau_2 . \qquad (11.26)$$

Подставим (11.23), (11.24), (11.25), (11.26) в (11.22), откуда

$$\tau_1 = \frac{T}{2} + \frac{v_o}{2U_o} - \sqrt{\frac{T^2}{4} - \frac{v_o^2}{4U_o^2} - \frac{r_o}{U_o} - \frac{v_o T}{2U_o}} . \qquad (11.27)$$

<u>Пример</u>.

Пусть в условиях примера, рассмотренного в 11.1.1, необходимо произвести сближение КА за 600 сек., минимизируя при этом расход топлива. Итак, r_o=60 км, v_o=40 м/сек, U_o=2 м/сек^2. Заданное время сближения (600 сек.) превышает минимальное время, равное 367.6 сек. (см. пример в 11.1.1).

Вид оптимального по расходу топлива управления соответствует управлению u(t), представленному на рис. 11.4. Произведем расчет характеристик управления.

Участок 1. Производится гашение начальной относительной скорости КА-инспектора. Длительность участка определяется в соответствии с формулой (11.27):

$$t_1 = \tau_1 = \frac{600}{2} + \frac{40}{2\cdot 2} - \sqrt{\frac{360000}{4} - \frac{1600}{4\cdot 4} - \frac{60000}{2} - \frac{24000}{2\cdot 2}} = 77.84$$

Через 77.84 сек. работы двигательной установки (u = -2 м/сек^2) объекты находятся на расстоянии, определяемом согласно (11.18),

$$x_1^1 = 60000 + 40 \cdot 77\,077.84 - \frac{2 \cdot 77.84^2}{2} = 57054.5 \text{ м}.$$

Объекты сближаются со скоростью, определяемой согласно (11.19),

x^1_2 = 40 - 2·77.84 = -115.68 м/сек.

Участок 2. Производится сближение объектов с постоянной скоростью, равной -115.68 м/сек, при выключенной двигательной установке. Длительность участка определяется согласно (11.25):

$$\tau_2 = 600 - 2 \cdot 77.84 + \frac{40}{2} = 464.32 \text{ сек.}.$$

Через 464.32 сек. объекты находятся на расстоянии, определяемом согласно (11.20),

x^2_1 = 57054.5 - 115.68·464.32 = 3342 м.

Участок 3. Включается двигательная установка для того, чтобы погасить скорость сближения КА. Длительность участка определяется согласно (11.24)

$$\tau_3 = 77.84 - \frac{40}{2} = 57.84 \text{ сек.}.$$

Через 57.84 сек. работы двигательной установки (u = 2 м/сек^2) расстояние между объектами определяется согласно (11.22) и равно

$$x_1^3 = 3342 - 115.68 \cdot 57.84 + \frac{2 \cdot 57.84^2}{2} \approx 0 \text{ м}.$$

Относительная скорость КА в конце участка определяется согласно (11.23), и она равна

x^3_2 = -115.68 + 2·57.84 = 0.

Таким образом, найдено управление, позволяющее произвести сближение КА за 600 сек. при минимальном расходовании топлива.

11.2. Алгоритм Ньютона в задаче поиска оптимального по быстродействию управления сближением КА

Принцип максимума Л.С. Понтрягина позволяет свести задачу оптимального управления к краевой задаче специального вида (см. 9.2). Будем рассматривать задачу Лагранжа с фиксированными концами. Особенностью такой задачи является то, что, если для фазовых переменных начальные условия заданы, и состояние может быть найдено для любого момента времени в результате решения задачи Коши, то для сопряженных переменных они не заданы. Произвольное задание начальных условий сопряженных переменных может привести к тому, что краевые условия на правом конце не будут выполняться. Рассогласование в выполнении таких краевых условий является функцией заданных начальных условий для сопряженных переменных. Метод Ньютона (см. 9.3) используется для решения трансцендентного уравнения, в результате чего невязки в краевых условиях на правом конце сводятся к нулю.

Применение метода Ньютона будем рассматривать на примере поиска оптимального по быстродействию управления в задаче сближения КА (11.1)-(11.5).

1. $J = \int_0^T dt \;\; \rightarrow \;\; \min .$

2. $\dot{x}_1 = x_2, \quad x_1(t) = x_{1o} + x_{2o}t + \dfrac{ut^2}{2},$

   $\dot{x}_2 = u, \quad x_2(t) = x_{2o} + ut .$

3. $u(t) \in G_u = [-U_o, +U_o], \;\; t \in (0,T].$

4. $x(0) = \| r_o \; v_o \|^T, \;\; x(T) = \| \; 0 \;\; 0 \; \|^T .$

Из решения сопряженной системы уравнений (см. 11.1.1) и рис.11.1 можно видеть, что момент переключения $t_п$ однозначно определяется начальными условиями для сопряженных переменных (ψ_{1o}, ψ_{2o}). Действительно,

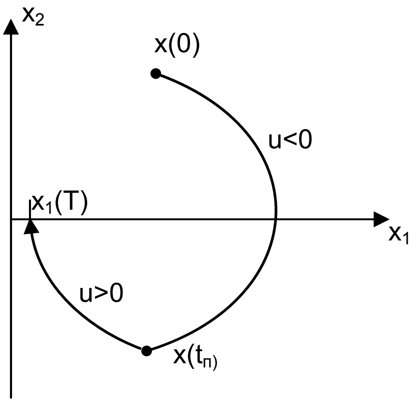
$\psi_2(t) = \psi_{2o} - \psi_{1o}\, t_п = 0,$

тогда $t_п = \psi_{2o} / \psi_{1o}$. Здесь ψ_{1o} = const, пусть $\psi_{1o} = 1$, тогда $t_п = \psi_{2o}$.

Рассмотрим, каким образом произвольное задание начальных условий для сопряженных переменных ($\alpha = \psi_{2o}$) влияет на выполнение краевых условий $x(T) = \| \; 0 \;\; 0 \; \|^T$. При этом в качестве невязки $d(\alpha)$ будем рассматривать значение

$d(\alpha) = x_1(T)$, при условии, что $x_2(T) = 0$.

Это можно пояснить следующим рисунком (11.6)



$\alpha = \psi_{2o} = t_п$
$d(\alpha) = x_1(T)$
$T = t_п + t_1$

Рис.11.6.

Найдем аналитическое выражение для $d(\alpha)$. Для этого рассмотрим движение системы по участкам

Участок 1: $(r_o, v_o) \xrightarrow{t_п} (x_1(t_п), x_2(t_п)), \quad u = -U_o$.

В соответствии с (11.7),(11.8)

$$x_1(t_п) = r_o + v_o t_п - \frac{U_o t_п^2}{2} \tag{11.28}$$

$$x_2(t_п) = v_o - U_o t_п \tag{11.29}$$

Участок 2: $(x_1(t_п), x_2(t_п)) \xrightarrow{t_1} (x_1(T), 0); u = +U_o$.

$$x_1(T) = x_1(t_п) + x_2(t_п) t_1 + \frac{U_o t_1^2}{2} = d(\alpha), \tag{11.30}$$

$$x_2(T) = x_2(t_п) + U_o t_1 = 0. \tag{11.31}$$

Подставляем (11.29) в (11.31), тогда

$$t_1 = t_п - v_o/U_o . \tag{11.32}$$

Подставляем (11.28),(11.29),(11.32) в (11.30), получаем

$$d(\alpha) = x_1(t_1) = -U_o t_п^2 + 2v_o t_п + r_o - \frac{v_o^2}{2U_o},$$

или, т.к. $\alpha = \psi_{2o} = t_п$

$$d(\alpha) = -U_o\alpha^2 + 2v_o\alpha + r_o - \frac{v_o^2}{2U_o} \quad . \qquad (11.33)$$

Тогда матрица частных производных $D(\alpha)$ (см.(9.13)) имеет вид

$$D(\alpha) = 2v_o - 2U_o\alpha. \qquad (11.34)$$

<u>Пример</u>.

Рассмотрим применение метода Ньютона в условиях примера, рассмотренного в 11.1.1. Тогда r_o = 60000 м; v_o = 40 м/сек; U_o = 2 м/сек^2.

Итерационный процесс начинается с задания произвольного α^o, и последующего уточнения его на основе формул (9.15):

$$\delta^k = -D^{-1}(\alpha^k)d(\alpha^k),$$

$$k = 0,1,....$$

$$\alpha^{k+1} = \alpha^k + \delta^k,$$

которое производится пока $\|\alpha^{k+1} - \alpha^k\|$ больше некоторого наперед заданного ε. Пусть $\varepsilon = 0.01$.

Подставляя заданные начальные условия в выражения (11.33), (11.34), получим

$d(\alpha) = -\alpha^2 + 80\alpha + 59600$,

$D(\alpha) = -4\alpha^2 + 80$.

Проведем итерационный процесс в соответствии с алгоритмом Ньютона (см. 9.3). Пусть $\alpha^o = 100$.

0. $\alpha^o = 100$; тогда
   $d(\alpha^o) = 47600$, $D(\alpha^o) = -320$, $\alpha^1 = \alpha^o - d(\alpha^o)/D(\alpha^o) = 248.75$.

1. $\alpha^1 = 248.75$; $\|\alpha^1 - \alpha^o\| > 0.01$;
   $d(\alpha^1) = -44253.125$; $D(\alpha^1) = -915$; $\alpha^2 = 200.39$.

2. $\alpha^2 = 200.39$; $\|\alpha^2 - \alpha^1\| > 0.01$;
   $d(\alpha^2) = -4681.1$; $D(\alpha) = -721.56$; $\alpha^3 = 193.9$.

3. $\alpha^3 = 193.90$; $\|\alpha^3 - \alpha^2\| > 0.01$;
   $d(\alpha^3) = -82.42$; $D(\alpha^3) = -695.6$; $\alpha^4 = 193.78$.

4. $\alpha^4 = 193.78$; $\|\alpha^4 - \alpha^3\| > 0.01$;
   $d(\alpha^4) = 1.02$; $D(\alpha^4) = -695.12$; $\alpha^5 = 193.78$.

5. $\alpha^5 = 193.78$; $\|\alpha^5 - \alpha^4\| < 0.01$ - конец.

Как отмечалось в 9.3. при произвольном задании α^o итерационный процесс может расходиться. Действительно, если положить, $\alpha^o = 300$, то $d(\alpha^o) = -96400$, что превышает начальное отклонение, равное 60000, т.е. процесс расходится.

11.3. Оптимальное многошаговое управление выполнением операции обслуживания КА

В 8.4.2 была поставлена задача управления выполнением операций обслуживания КА. Рассмотрим такую задачу для одной операции обслуживания в следующем виде

1. Функционал

$$\int_0^T (x^2(t) + u^2(t))dt \to \min ,$$

2. Дифференциальные связи

$$\dot{x}(t) = \varepsilon(t)\ u(t),$$

3. Ограничения вдоль траектории:
   $u(t) \in [-U_o , U_o]$;
   $x(t) \le x^{зад}$.
4. Краевые условия: $x(0) = 0$; $x(T) = x^{зад}$.

Здесь $x^{зад}$ - заданное состояние операции обслуживания, характеризующее факт ее выполнения;

$\varepsilon(t)$ - заданная функция, принимающая значение 1 в моменты времени, когда КА попадает в зону видимости того или иного пункта, и 0 в противном случае.

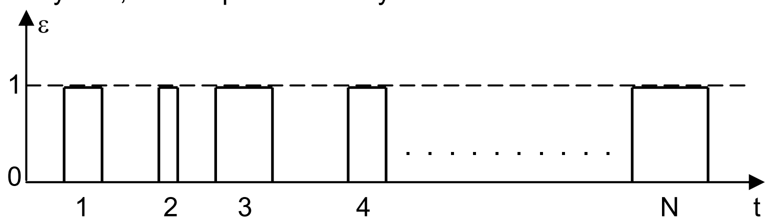


Рис. 11.7

Проведем дискретизацию данной модели, для чего дифференциальные связи перепишем в виде (см. 8.7)

$$\frac{x(t+\delta t)-x(t)}{\delta t}=\varepsilon(t)u(t)$$

отсюда

$$x(t + \delta t) - x(t) = \varepsilon(t)\ u(t)\ \delta t$$

или, нумеруя интервалы, когда операция обслуживания может выполняться, получим

$$x(k+1) = x(k) + u(k). \tag{11.35}$$

Функционал будет иметь вид

$$J=\sum_{k=0}^{N-1}\ (x^2(k)+u^2(k))\to \min . \tag{11.36}$$

Ограничения вдоль траектории:

$u(k) \in [-U_o,+U_o]$;

$x(k) \le x^{зад}$;   k= 0,...,N-1.

Краевые условия: $x(0) = 0$; $x(N) = x^{зад}$.

Для решения такой задачи может использоваться алгоритм динамического программирования, основанный на принципе оптимальности Р. Беллмана (см. 10.4) и позволяющий получать управление в форме синтеза.

Пусть $N = 5$; $U_o = 4$; $x^{зад} = 6$. Применим алгоритм динамического программирования для решения данной задачи. Функция Беллмана (см.(10.12)) будет иметь вид

$$S_{N-h}\ (x(N-h))=\min_{u(N-h)\in[-4,4]}\ (x^2(N-h)+u^2(N-h)+S_{N-h+1}(x(N-h+1))).$$

Вычисления начинаются с предпоследнего шага.

1). h = 1; N - h = 4; x(5) = 6.

$$S_4(x(4))=\min_u (x^2+u^2+S_5(x(5)))$$

С учетом ограничений на управление $u(4)\in[-4,4]$ заданное состояние $(x(5)=6)$ может быть достигнуто, если $x(4) \in \{2,3,4,5,6\}$. Тогда, учитывая, что $S_5(x(5))= 0$, получим

$S_4(2) = 2^2 + 4^2 + 0 = 20$

$S_4(3) = 3^2 + 3^2 + 0 = 18$

$S_4(4) = 4^2 + 2^2 + 0 = 20$

$S_4(5) = 5^2 + 1^2 + 0 = 26$

$S_4(6) = 6^2 + 0^2 + 0 = 36$

Результаты расчета значений функции Беллмана и возможные переходы системы из состояния в состояние представлены на следующем рисунке.

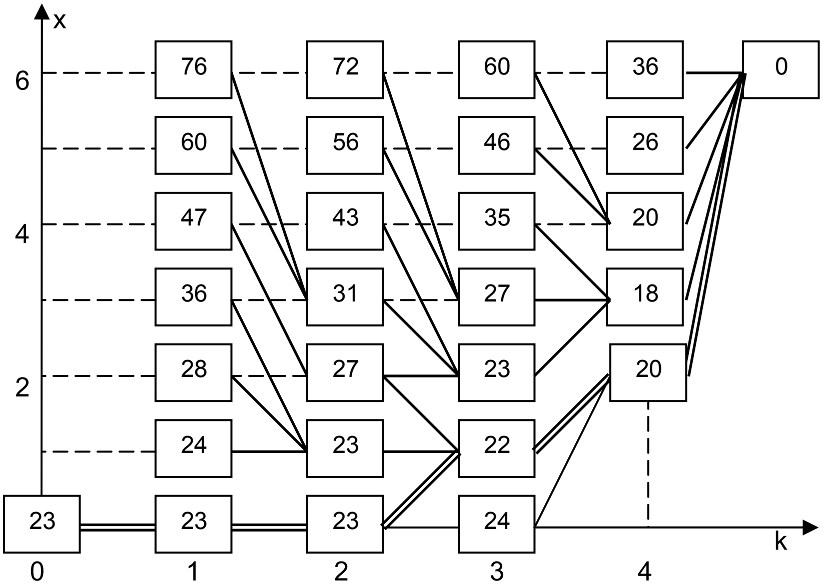


Рис.11.8.

2). h = 2; N - h = 3.

$$S_3(x(3)) = \min_u (x^2 + u^2 + S_4(x(4)))$$

Пусть x(3) = 6. Тогда

$$S_3(6) = \min \begin{cases} 36 + 0 + 36 = 72 \\ 36 + 1 + 26 = 63 \\ 36 + 4 + 20 = 60 \leftarrow \min, u = -2 \\ 36 + 9 + 18 = 63 \\ 36 + 16 + 20 = 72 \end{cases}$$

Далее x(3) = 5;

$$S_3(5) = \min\begin{cases} 25+1+36=62 \\ 25+0+26=51 \\ 25+1+20=46 \leftarrow \min, u=-1 \\ 25+4+18=47 \\ \dots\dots\dots\dots\dots\dots\dots \end{cases}$$

Теперь x(3) = 4;

$$S_3(4) = \min\begin{cases} \dots\dots\dots\dots\dots\dots\dots \\ 16+0+20=36 \\ 16+1+18=35 \leftarrow \min, u=-1 \\ 16+4+20=40 \\ \dots\dots\dots\dots\dots\dots\dots \end{cases}$$

x(3) = 3;

$$S_3(3) = \min\begin{cases} \dots\dots\dots\dots\dots\dots\dots \\ 9+0+18=27 \leftarrow \min,\ u=0 \\ 9+1+20=30 \end{cases}$$

Дальнейшие вычисления производим аналогично предыдущим.

$$S_3(2) = \min\begin{cases} \dots\dots\dots\dots\dots\dots\dots \\ 4+1+18=23 \leftarrow \min,\ u=+1 \\ 4+0+20=24 \end{cases}$$

$$S_3(1) = \min\begin{cases} \dots\dots\dots\dots\dots\dots\dots \\ 1+4+18=23 \\ 1+1=20=22 \leftarrow \min,\ u=+1 \end{cases}$$

$$S_3(0) = \min\begin{cases} \dots\dots\dots\dots\dots\dots\dots \\ 0+9+18=27 \\ 0+4+20=24 \leftarrow \min,\ u=+2 \end{cases}$$

3). h = 3; N-h = 2.

$$S_2(6) = \min\begin{cases} 36+0+60=96 \\ 36+1+46=83 \\ 36+4+35=75 \\ 36+9+27=72 \leftarrow \min,\ u=-3 \\ 36+16+23=75 \end{cases}$$

$$S_2(5)=\min\begin{cases}\text{..........................}\\ 25+1+35=61\\ 25+4+27=56\leftarrow \min, u=-2\\ 25+9+23=57\\ \text{...........................}\end{cases}$$

$$S_2(4)=\min\begin{cases}\text{..........................}\\ 16+1+27=44\\ 16+4+23=43\leftarrow \min, u=-2\\ 16+9+22=47\\ \text{...........................}\end{cases}$$

$$S_2(3)=\min\begin{cases}\text{.......... ....}\\ 9+1+23=33\leftarrow \min,\ u=-1\\ 9+4+22=35\\ 9+9+24=42\end{cases}$$

$$S_2(2)=\min\begin{cases}\text{.......... ....}\\ 4+0+23=27\leftarrow \min,\ u=0\\ 4+1+22=27\leftarrow \min,\ u=-1\end{cases}$$

$$S_2(1)=\min\begin{cases}\text{.......... ....}\\ 1+0+22=23\leftarrow \min,\ u=0\\ 1+1+24=26\end{cases}$$

$$S_2(0)=\min\begin{cases}\text{.......... ....}\\ 0+1+22=23\leftarrow \min,\ u=+1\\ 0+0+24=24\end{cases}$$

4). h = 4; N-h = 1.

$$S_1(6)=\min\begin{cases}36+0+72=108\\ 36+1+56=93\\ 36+4+43=83\\ 36+9+31=76\leftarrow \min,\ u=-3\\ 36+16+27=79\end{cases}$$

$$S_1(5)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 25+0+56=81\\ 25+1+43=69\\ 25+4+31=60\leftarrow \min,\ u=-2\\ 25+9+27=61\end{cases}$$

$$S_1(4)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 16+0+43=59\\ 16+1+31=48\\ 16+4+27=47\leftarrow \min,\ u=-2\\ 16+9+23=48\end{cases}$$

$$S_1(3)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 9+0+31=40\\ 9+1+27=37\\ 9+4+23=36\leftarrow \min,\ u=-2\\ 9+9+23=41\end{cases}$$

$$S_1(2)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 4+0+27=31\\ 4+1+23=28\leftarrow \min,\ u=-1\\ 4+4+23=31\end{cases}$$

$$S_1(1)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 1+0+23=24\leftarrow \min,\ u=0\\ 1+1+23=35\end{cases}$$

$$S_1(0)=\min\begin{cases}\ldots\ldots\ldots\ \ldots\ldots\ldots\ \ldots\\ 0+1+23=24\\ 0+0+23=23\leftarrow \min,\ u=0\end{cases}$$

5). h = 5; N-h = 0. Исходное состояние x(0) = 0 - задано, тогда рассчитывается значение

$$S_0(0)=\min\begin{cases}\ldots\ldots\ldots\ldots\ldots\ldots\\ 0+1+24=25\\ 0+0+23=23\leftarrow \min, u=0\end{cases}$$

Таким образом, осуществляя движение от предпоследнего шага к начальному шагу, производится вычисление управления, как функции возможного состояния u(k,x(k)). На начальном шаге при известном исходном состоянии x(0) и управлении u(0) можно вычислить, используя конечноразностные уравнения (.), состояние x(1), для которого уже известно оптимальное управление u(1), и т.д. В этом случае оптимальное управление имеет вид

u(0)=0; u(1)=0; u(2)=+1; u(3)=+1; u(4)=+4.

Оптимальное значение функционала равно 23.

Из рис.11.8 видно, что в каком бы состоянии ни была операция управления, имеется оптимальное управление из этого состояния. Таким образом, управление найдено в форме синтеза.

РАЗДЕЛ 3. ПРИНЯТИЕ РЕШЕНИЙ В СЛОЖНЫХ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМАХ

12. ПОСТАНОВКА ЗАДАЧ ПОДГОТОВКИ И ПРИНЯТИЯ РЕШЕНИЙ В СЛОЖНЫХ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМАХ

12.1. Понятие сложных военно-технических систем и особенности принятия решений в них.

Современные военно-технические системы (комплексы), являющиеся основным элементом Вооруженных Сил любой страны, прежде всего, предназначены для решения боевых задач в ходе подготовки и ведения военных действий, которые характеризуются антагонизмом и динамичностью целей, стоящих перед конфликтующими сторонами, противоречивой обстановкой, складывающейся в процессе боевых действий. При этом сложность военно-технических систем проявляется, прежде всего, в таких аспектах, как: структурная сложность; сложность функционирования; сложность выбора поведения; сложность развития. Современная наука создала богатый аппарат, позволяющий преодолевать трудности, вызванные воздействием факторов сложности. Это аппарат системных (системно-кибернетических) исследований, центральное место в котором занимает системный (комплексный) подход, широко используемый в настоящее время при решении крупномасштабных проблем, связанных с созданием и применением сложных и больших систем (комплексов).

Системы управления космическими силами и средствами различного целевого назначения так же представляют собой сложные военно-технические системы, имеющие, как правило, разветвленную иерархическую структуру, в состав которых входят управляющие подсистемы (людские коллективы) различных уровней, и технические средства управления. Принятие решений в таких системах должно базироваться на моделях, отражающих специфику решаемых управленческих задач, а также разнородные требования, предъявляемые к решению, в различных условиях обстановки. В этой связи целесообразно определить, какими существенными особенностями обладают сложные системы по сравнению с системами простыми.

В современной научно-технической литературе отсутствуют общепринятые определения с л о ж н о й системы. В ряде случаев наряду с понятием сложной системы широко используется понятие б о л ь ш о й системы. Вместе с тем можно отметить, что с точки зрения решения проблем, связанных с принятием решений, эти отличия не являются существенными. Анализ многочисленных публикаций по данной тематике показывает, что в настоящее время можно различать три основных подхода к определению понятия сложной (большой) системы:

а) словесный (вербальный) подход на уровне перечисления признаков;

б) словесный (вербальный) подход на уровне формулирования обобщенного понятия;

в) количественный или математический подход.

Рассмотрим определения сложных систем, данные в рамках этих подходов.

Система называется с л о ж н о й, если она обладает хотя бы одним из следующих признаков:

1) система функционирует в условиях существенной неопределенности воздействия среды на ее поведение;

2) система осуществляет целенаправленный (многоцелевой) выбор своего поведения, или поведение системы удовлетворять множеству противоречивых требований;

3) система допускает разбиение на подсистемы, изучение которых с учетом влияния других подсистем носит содержательный характер;

4) наличие в составе системы коллективов людей. Приведенное определение сложной системы относится к вербальному описанию понятия на уровне перечисления признаков.

У.Р.Эшби предпринял попытку дать сжатую формулировку на уровне формулирования обобщенного понятия, отображающую концепцию относительности понятия сложности системы.

Система является с л о ж н о й, когда: задан определенный наблюдатель, с определенными средствами и методикой и система, которая в каких-либо целях исследования слишком велика для него, например, он не может наблюдать ее полностью, или выполнять все вычисления, необходимые для предсказания ее поведения.

Тогда *т а к а я* система, для *т а к о г о* исследователя в *д а н н ы х* целях исследования является сложной.

В качестве свойств сложных систем У.Р.Эшби отмечает два наиболее характерных:

- *э м е р д ж е н т н о с т ь* (заключается в наличии у сложной системы таких свойств, которые не выводятся непосредственно на основании изучения свойств ее отдельных элементов и подсистем);
- *з н а ч и т е л ь н о е р а з н о о б р а з и е*, действующих на систему возмущений и соответствующее разнообразие возможных состояний, что приводит к необходимости наличия необходимого разнообразия управляющих воздействий для эффективного управления сложной системой.

Современные военно-технические системы представляют собой сложные системы, так как

- представляют собой совокупность технических средств, обеспечивающих выполнение целевых задач, и людских коллективов, управляющих техническими средствами;
- выполнение целевых задач происходит в условиях существенной неопределенности, которая связана со случайными сбоями в работе техники и ошибочными непрогнозируемыми действиями личного состава;
- при управлении военно-техническими системами к принимаемым решениям предъявляется множество противоречивых требований, связанных с многоцелевым характером функционирования системы;
- система управления военно-техническими системами, как правило, имеет сложную иерархическую структуру, на различных уровнях которой решаются специфические задачи принятия решений, связанные между собой объектом управления.

В соответствии с особенностями сложных систем и с теми признаками, которые характеризуют систему как сложную, можно сформулировать особенности моделей принятия решений в сложных системах.

1). Принятие решений в сложных системах представляет собой многомодельное исследование, необходимым признаком которого является разработка системы взаимосвязанных моделей, отражающих различные аспекты поведения системы и особенности принятия решения в них.

2). Модели принятия решений должны строиться с учетом факторов неопределенности воздействия внешней среды.

3). Модели принятия решений должны учитывать разнородные противоречивые требования, предъявляемые к системе. В результате этого модель становится многокритериальной (с векторным критерием оптимальности).

4). Сложная военно-техническая система имеет, как правило, разветвленную иерархическую структуру. В такой системе осуществляется взаимосвязанное принятие решение в подсистемах различных иерархических уровней. Тогда модели принятия решений должны строиться на основе использования методов декомпозиции и координации.

Методологическую основу рассмотрения проблем принятия решений в сложных военно-технических системах составляет системный (комплексный) подход, в рамках которого все многообразие существующих объектов рассматривается как объединение элементов, находящихся в определенных отношениях друг к другу и выступающих как единое целое по отношению к внешней среде. Можно выделить две основные функции системного подхода: а) постановка проблем, связанных с получением новых научных знаний или решением новых задач; б) методологический анализ существующего знания.

Анализ применений методологии современной теории управления, основанной на системном подходе, к сложным военно-техническим системам можно заключить, что она включает в себя две основные части:

1) методологию системного анализа, базирующуюся на логико-эвристической и отчасти математической основе;

2) методологию современной теории выбора, опирающуюся на хорошо развитый математический аппарат.

Обе эти части взаимно проникают и обогащают друг друга, что необходимо учитывать при постановке и решении задач принятия решений. При этом необходимо учитывать, что при принятии решения в сложных военно-технических системах существенную роль играет л и ц о, п р и н и м а ю щ е е р е ш е н и е (ЛПР). Это, как правило, командир, несущий ответственность за принятые решения, при взаимодействии с которым на основе использования методов системного анализа ставятся и доопределяются задачи выбора.

С целью рассмотрения на единой концептуальной основе многочисленных задач принятия решений в сложных системах целесообразно использовать единую обобщенную постановку задач принятия решений.

12.2. Обобщенная постановка задачи принятия решений в сложных системах.

Требование достижения соответствия математического описания проблемы принятия решений обстановке, в которой это решение реализуется, обусловливает необходимость использования общей методологической основы для рассмотрения различных задач принятия решений в сложных военно-технических системах, проведения соответствующих классификаций, установления связей между задачами различных классов. С этой целью целесообразно использовать структурно-математический подход [40] и обобщенную постановку задачи выбора.

Математическая постановка задачи выбора (принятия решений) в общем виде может быть представлена как:

$$(S, K), \tag{12.1}$$

где

$S = \{ S_k, k \in Z \}$ - множество моделей, отражающих различные аспекты принятия решений в сложных системах;

K - правила согласования (координации) решений, полученных на моделях из S.

Отдельная модель принятия решений S_k была рассмотрена в 1.4. В соответствии с (1.1) ее можно определить как

$$S_k = (Q(s), \Delta, \{ r_i, i \in C \}, \{ f_j, j \in G \}) . \tag{12.2}$$

Здесь

Q(s) - исходная структура выбора (модель) типа s;

Δ - пространство альтернатив (решений);

$\{r_i, i \in C\}$ - множество отношений, ограничивающих выбор;

$\{f_j, j \in G\}$ - множество отношений предпочтения.

Задачу принятия решения (12.2) кратко можно характеризовать парой

$$(\Delta, f) , \tag{12.3}$$

здесь Δ - множество допустимых альтернатив;

f - правило (критерий) выбора решения.

При формулировании понятия "сложная система" было определено, что система является сложной, если она обладает, в частности, одним из следующих свойств:

1) система функционирует в условиях существенной неопределенности воздействия среды;

2) в системе осуществляется целенаправленный (многоцелевой) выбор ее поведения, или поведение системы удовлетворять множеству противоречивых требований;

3) система допускает разбиение на подсистемы, изучение которых с учетом влияния других подсистем носит содержательный характер.

В соответствии с этими особенностями необходимо строить и конкретные задачи принятия решений в сложных системах. Основным при этом является: учет факторов неопределенности обстановки, в которой, как правило, функционирует сложная система; учет разнородных требований, предъявляемых к оптимальному решению; учет особенностей распределенного принятия решений в сложных системах, обладающих иерархической структурой.

1. Факторы, связанные с неопределенностью обстановки.

Основной способ исследования сложных систем в системном анализе - это построение модели. В общем виде модель системы задается как отношение (или система отношений) на образующих множествах.

$$S \subseteq X_1 \times X_2 \times \cdots \times X_n \times \Omega \ .$$

Здесь Ω - множество состояний среды.

Характерной особенностью данного описания S, отличающего его от соответствующего описания системы в условиях полной определенности, является наличие множества Ω, от элементов которого $\omega \in \Omega$ зависят все или часть компонент, описывающих систему. Тогда неопределенность знаний о среде учитывается при задании множества альтернатив как

$$\Delta\,(\omega) \subseteq \Delta \times \Omega \ .$$

Здесь ω - элемент неопределенности, а Ω - множество неопределенных состояний среды; $\omega \in \Omega$.

Задача принятия решения (12.3) может быть представлена в виде

$$(\Delta\,(\omega),\ f(\omega)), \quad \omega \in \Omega. \qquad (12.4)$$

В зависимости от степени полноты имеющихся о среде знаний различают следующие типы среды с неопределенностью: стохастическая, целенаправленная и неизвестная. Для каждого типа среды используется та или иная форма ее описания, при этом на Ω вводится соответствующая математическая структура. Так, в случае, если среда статистически устойчива (т.е. можно количественно описывать частоту появления тех или иных состояний), вводится вероятностная математическая структура; в случае, если среда целенаправленная (известна цель воздействия среды на систему), используются конфликтные (игровые) методы. Для описания влияния неизвестной среды используются специфические способы описания предположений лица, принимающего решение, о влиянии среды, основанные, например, на "нечетких" множествах.

2. *Учет разнородных требований, предъявляемых к оптимальному решению.*

При создании, исследовании, применении и развитии сложных военно-технических систем всесторонняя оценка качества принимаемых решений становится возможной только при использовании нескольких показателей (нескольких целевых, критериальных функций). Это приводит к появлению в задачах выбора нового вида неопределенности - критериальной неопределенности. Следует отметить, что отношение предпочтения как правило является математической моделью реально существующего предпочтения лица, принимающего решения, не альтернативных объектов в целом, а лишь определенных с в о й с т в этих объектов. Обычно таких свойств несколько, и, следовательно, адекватное представление проблемы выбора требует введения нескольких отношений предпочтения, учитывающих всю совокупность этих свойств. Такие задачи часто называют задачами многокритериальной (векторной) оптимизации. Анализ задач принятия решений, формулируемых в этих условиях, когда даже отсутствует влияние неопределенных факторов, связанных с воздействием на систему внешней среды, содержит специфическую неопределенность, проявляющуюся в необходимости согласования различных целевых функций с помощью построения результирующих отношений предпочтения.

С учетом разнородных требований, предъявляемых к оптимальному решению, задачу (12.3) можно переписать в виде

$$(\Delta, \{ f_j, j \in G \}) . \qquad (12.5)$$

Главной особенностью задач многокритериального выбора является то, что данные задачи не являются корректными в рамках аксиоматики, принятой в классической теории оптимизации и принятия решения. Некорректность задач многокритериального выбора обуславливает необходимость использования для ее решения специальных методов, основанных на регуляризации (доопределении) задачи путем привлечения от лица, принимающего решение, дополнительной качественной и количественной информации о свойствах оптимальных решений, которая может выражаться в некоторых аксиомах, принципах оптимальности, и т.п.

3. Принятие решений в иерархических системах.

Большинство современных военно-технических систем относится к классу, так называемых, больших систем или систем крупного масштаба. Развитие систем управления в них связано с ростом сложности и масштабности решаемых целевых задач. Существенной особенностью такого развития является специализация функций, которая приводит к тому, что существующая система управления разбивается на совокупность подсистем, решающих специализированные задачи, которые имеют информационную, методическую и алгоритмическую общность. Это сопровождается децентрализацией процесса обработки информации и, как следствие этого, децентрализацией принятия решения. Действительно, динамичность обстановки и связанный с этим рост потоков информации может привести к тому, что полностью децентрализованный сбор и обработка информации либо технически невозможны, либо приводят к значительному запаздыванию в принятии решения. Указанные особенности требуют разбиения исходной системы на совокупность связанных, но самостоятельно функционирующих подсистем.

Появление в системе отдельных звеньев, способных с необходимой оперативностью перерабатывать всю поступающую информацию и принимать решения в рамках своей компетенции означает, по существу, появление в системе иерархической структуры, при которой ряд подсистем подвергается декомпозиции, а процесс управления децентрализации. Однако такое разбиение исходной системы служит источником н о в о й н е о п р е д е л е н н о с т и. Действительно, каждая j-я подсистема принимает решение в соответствии со своими собственными целями f_j, не тождественными в общем случае целям других подсистем и системы в целом. В этой ситуации возникает задача с о г л а с о в а н и я (координации) решений подсистем, и одна из подсистем высшего иерархического уровня (Центр) наделяется специальными полномочиями по решению этой задачи. По существу, координационная задача служит для учета эмерджентных свойств системы, подвергнутой декомпозиции.

Задача выбора (12.3) в этом случае имеет свои специфические особенности. Действительно, пространство альтернатив Δ в этом случае представляет собой $\Delta = \Delta^o \times \Delta^1 \times \cdots \times \Delta^p$, где Δ^o - пространство альтернатив Центра, а Δ^j, j=1,...,p - пространство альтернатив j-ой подсистемы; соответственно $f = (f_o, f_1, \ldots \ldots, f_p)$. Тогда общая задача выбора распадается на совокупность связанных задач:

- задачу центра (координатора) $S_o = (\Delta^o \times P, f_o)$,
- задачи подсистем (локальные) $S_j = (P \times \Delta^j, f_j)$,

здесь P - множество связующих (координирующих) сигналов.

Различные способы организации обмена информацией между координатором и подсистемами, а также различные способы формирования и учета координирующих сигналов порождают множество различных алгоритмов декомпозиционного (координационного) принятия решений в иерархических военно-технических системах.

12.3. Автоматизированная система управления КА, как сложная военно-техническая система

Интенсивное освоение космического пространства в мирных и военных целях привело к тому, что в космосе одновременно находятся десятки и даже сотни космических аппаратов, решающих различные задачи в процессе орбитального полета. Это прежде всего космические аппараты навигации, связи, разведки, КА, решающие метеорологические, геодезические и другие задачи. Для решения некоторых задач (например, навигации или разведки) однотипные КА (а в ряде случаев и разнотипные КА) могут объединяться в орбитальные системы (ОС). Совокупность указанных КА и орбитальных систем образует орбитальную группировку, которой необходимо постоянно управлять при реализации целевого назначения космических аппаратов. Для решения задач, связанных с управлением КА, осуществляется:

- сбор информации о состоянии движения КА (измерение текущих навигационных параметров), необходимой для поддержания заданной баллистической структуры системы КА;
- сбор информации о состоянии бортовой специальной и обеспечивающей аппаратуры КА (телеметрические измерения), необходимой для оценки возможностей технических средств КА;
- сбор информации о состоянии выполнения целевых задач КА.

По результатам анализа полученной информации вырабатываются управляющие воздействия, которые передаются на борт КА в форме разовых команд или временных программ работы аппаратуры. Для решения столь широкого круга задач, связанных с управлением КА, создается автоматизированная система - АСУ КА. Структура АСУ КА приведена на рис. 12.1.

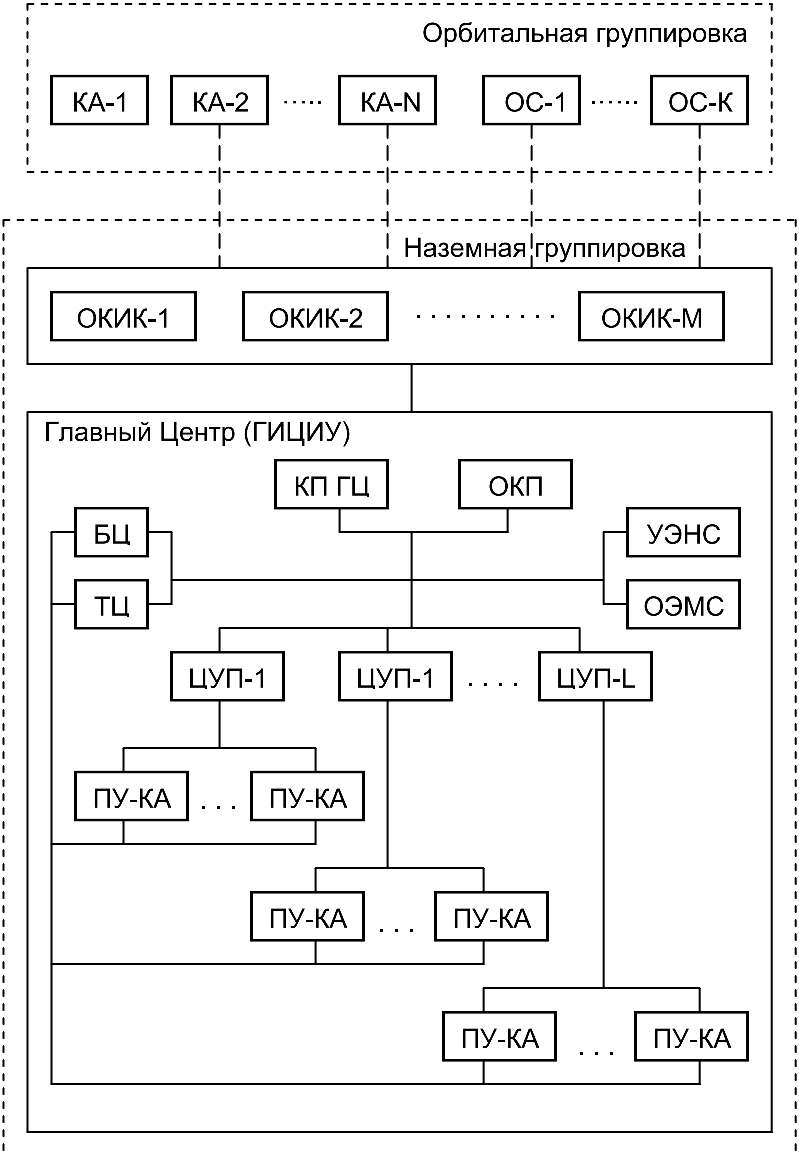

Орбитальная группировка
КА-1
КА-2
КА-N
ОС-1
ОС-К
Наземная группировка
ОКИК-1
ОКИК-2
ОКИК-М
Главный Центр (ГИЦИУ)
КП ГЦ
ОКП
БЦ
ТЦ
УЭНС
ОЭМС
ЦУП-1
ЦУП-1
ЦУП-L
ПУ-КА
ПУ-КА
ПУ-КА
ПУ-КА
ПУ-КА
ПУ-КА

Рис.12.1

С целью повышения потенциальных пространственно-временных возможностей управления КА в состав АСУ КА входит совокупность территориально распределенных отдельных командно-измерительных комплексов (ОКИК), объединяющих расположенные в одном географическом районе радиотехнические средства и средства обработки информации, которые предназначены для непосредственного приема с КА и передачи на КА всей измерительной и управляющей информации. ОКИК по каналам связи и передачи данных обменивается информацией с Главным Центром (Главный центр испытаний и управления - ГИЦИУ).

В состав Главного Центра входят: командный пункт главного центра (КП ГЦ), осуществляющий общее оперативное руководство дежурными сменами подразделений Главного центра; отдел координации и планирования (ОКП), обеспечивающий согласование планов работы подразделений, управляющих КА; управление эксплуатации наземных средств (УЭНС), ответственное за состояние средств наземной группировки (готовность к работе, отказ, техническое обслуживание, ремонт, и т.д.); отдел электромагнитной совместимости радиоэлектронных средств (ОЭМС), предоставляющий информацию о влиянии радиопомех (как создаваемых техническими средствами наземной группировки, так и посторонних) на работу наземных средств; баллистический центр (БЦ), предоставляющий информацию о состоянии движения КА; телеметрический центр (ТЦ), анализирующий состояние бортовых систем КА по результатам телеизмерений.

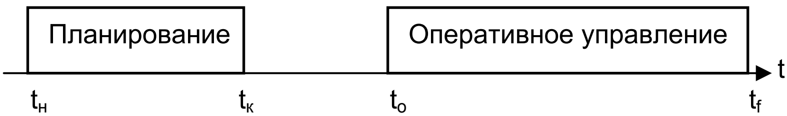
Основными подразделениями ГИЦИУ, ответственными за непосредственное управление КА, являются центры управления полетами (ЦУП), в состав которых входят пункт управления КА (ПУ КА). Именно на пунктах управления КА и осуществляется управление конкретными космическими аппаратами, для чего решаются задачи планирования (перспективного, долгосрочного, оперативного), а также задачи оперативного управления. Управление КА осуществляется в соответствии с технологическими циклами, при этом в качестве основной функции управления КА рассматривается функция принятия (выбора) решения.

Особенности функционирования АСУ КА, как сложной военно-технической системы, влияют на математические формализмы, которые следует использовать при принятии решений. Ранее были рассмотрены особенности моделей сложных военно-технических систем, конкретизируем их применительно к АСУ КА.

1). Система функционирует в условиях существенной неопределенности обстановки. При управлении КА решаются две основные группы задач.

Первая группа задач связана с предварительным планированием действий, обеспечивающих управление КА, таких как измерение текущих навигационных параметров КА, оценка технического состояния КА по результатам телеизмерений, выдача на борт КА разовых команд и временных программ и др.

Вторая группа задач обеспечивает контроль выполнения предварительно разработанной программы управления КА - плана (данный этап часто называют оперативным управлением). Решение указанных задач разнесено во времени.



Планирование осуществляется на основе информации, полученной к моменту $t_н$, а реализация плана (программы управления) начинается в момент $t_о$. На интервале времени $(t_н,t_о]$ на систему воздействуют возмущения различного рода. Часть из них, такие, например, как выход из строя технических устройств в результате неисправностей, может быть спрогнозирована, другие, например, такие, как ошибки операторов, неквалифицированные действия личного состава, статистической оценке и прогнозу не подвергаются. Совершенно своеобразными возмущениями, характерными для военных систем, являются воздействия противника. В этих условиях предварительно разработанные планы могут быть не реализуемы в изменившейся обстановке, что является фактором, усложняющим принятие решений на различных этапах управления в АСУ КА.

2). Система имеет выраженный многоцелевой характер функционирования. Прежде всего, здесь можно различать две большие группы взаимосвязанных целей: цели объекта управления - орбитальной группировки КА; цели наземной группировки, обеспечивающей управление орбитальной группировкой и включающей в свой состав совокупность технических средств управления. Орбитальная группировка состоит из КА различного целевого назначения, управление которыми осуществляется в соответствии со специфической технологией. В этой связи технология управления КА и его предназначение, естественно, должны учитываться в моделях принятия решений. Анализ особенностей функционирования наземных средств управления показывает, что при планировании их работы целесообразно повышать качество проведения сеансов управления, минимизировать расход ресурсов, привлекаемых к управлению, и т.д. Стремление повысить эффективность принимаемых в АСУ КА решений приводит к тому, что в моделях необходимо учитывать множественность целей, что усложняет данные модели и алгоритмы поиска рациональных решений.

3). Структура системы имеет сложное многоуровневое иерархическое строение. Непосредственное планирование и управление целевым применением КА осуществляется на пунктах управления ЦУП-ов. При этом ряд КА решает свои задачи совместно с другими КА в составе орбитальной системы аппаратов, тогда результаты независимого планирования работы отдельного КА должны согласовываться (координироваться) на уровне системы КА. Важной особенностью АСУ КА является то, что наземные средства управления, каналы связи, вычислительные средства, используемые для обработки информации, являются общими для КА различных типов. Таким образом, программы управления операциями отдельных КА должны координироваться: на уровне системы КА (на ПУ КА); на уровне ЦУПа, планирующего управление разнотипных КА, входящих в одну орбитальную систему; на уровне всей орбитальной группировки КА в целом (в ОКП). Такая иерархическая структура управления КА должна сопровождаться использованием специфических (координационных) моделей и алгоритмов принятия решений.

12.4. Модели принятия решений в АСУ КА

Одно из центральных мест при концептуальном описании процессов управления в АСУ КА занимает понятие "о п е р а ц и я", под которой обычно понимается действие или система действий, объединенных общим замыслом и единой целью. Вследствие этого цель функционирования каждого отдельного КА реализуется в ходе выполнения им операций, связанных с информационным, вещественным и энергетическим обменом с пунктами управления (обслуживания), а также с другими КА. Понятие операции является основным системообразующим понятием, позволяющим объединить описание различных видов деятельности КА (движение, работу аппаратуры, расход ресурса и т.п.). Содержание и специфика каждой выполняемой операции находит свое отражение в задании соответствующих параметров, характеризующих:

- результаты выполнения операций (объем, качество, время выполнения операций и т.п.);
- расход ресурсов при выполнении операций;
- информационные и материальные потоки, возникающие в ходе выполнения операций.

Состояние КА в фиксированный момент времени можно характеризовать совокупностью значений параметров операций, выполняемых КА в указанный момент времени. В целом же процесс функционирования КА можно интерпретировать как процесс выполнения им комплекса операций (работ), связанных с переводом КА из одного фиксированного состояния в другое фиксированное состояние. Аналогичный вывод оказывается справедлив и при моделировании функционирования элементов и подсистем системы автоматизированного управления (например, наземного комплекса управления - НКУ), при моделировании функционирования АСУ КА в целом. Таким образом, используя концепцию комплексов операций, удается связать в единое целое все основные аспекты функционирования АСУ КА, связанные с вещественным, энергетическим, информационным обменом ее основных элементов и подсистем друг с другом и окружающей средой.

На основе концептуального описания процессов автоматизированного управления КА проведем построение двух конкретных классов математических моделей, используемых для планирования работы средств управления.

12.4.1. Характеристика ситуации управления КА

Первым шагом построения моделей планирования операций, связанных с управлением КА (операций управления), является проведение процедуры структуризации - выяснение того, что же является здесь решением (альтернативой), какова его структура, из каких компонент оно состоит и чем определяется. Для этого необходимо провести формализованное описание ситуации управления. Для описания ситуации управления КА введем следующие базисные множества.

$A = \{A_\nu\}$, $\nu \in N = \{1,..., n\}$ - множество космических аппаратов (КА);

$B = \{B_\mu\}$, $\mu \in M = \{1,..., m\}$ - множество пунктов управления (отдельных командно-измерительных комплексов - ОКИК);

$D = \{D_æ\}$, $æ \in K = \{1,...,к\}$ - множество видов работ, проводимых с КА;

$C = \{C_\lambda\}$. $\lambda \in L = \{1,...,l\}$ - множество средств управления;

$T = (t_o, t_f]$ - множество моментов времени (интервал планирования операций).

Тогда множество работ (операций), которые необходимо запланировать, можно характеризовать как $A \times D = \{<\nu, æ>\}$.

Обозначим через D^ν - множество работ, проводимых с ν-м КА, $D^\nu = \{D^\nu{}_æ\}$, $æ \in K^\nu = \{1,..., к^\nu\}$, и C^μ- множество средств управления, размещенных на μ-м пункте, $C^\mu = \{C^\mu{}_\lambda\}$, $\lambda \in L^\mu = \{1,..., l^\mu\}$. Тогда $\cup D^\nu = D$, $\nu \in N$; $\cup C^\mu = C$, $\mu \in M$.

Динамика относительного перемещения системы КА-ОКИК может задаваться функцией (отображением) вида

$$\varepsilon: A \times B \times T \to \{0, 1\}, \tag{12.6}$$

которое называют [39] к о н т а к т н ы м потенциалом. При этом

$\varepsilon(\nu,\mu,t) = 1$, если A_ν находится в зоне обслуживания B_μ в момент t;

$\varepsilon(\nu,\mu,t) = 0$, в противном случае.

Так как множества A и B конечны, то функцию $\varepsilon(\nu,\mu,t)$ можно представить матричной функцией $E(t) = \|\varepsilon_{\nu\mu}(t)\|$, где $\varepsilon_{\nu\mu}(t) = \varepsilon(\nu,\mu,t)$.

Оснащенность ОКИК различными средствами, а также возможности средств по выполнению различных работ с КА можно характеризовать функциональным отношением (отображением)

$$\theta: A \times D \times B \times C \to \{0, 1\}, \tag{12.7}$$

которое называют [39] потенциалом д о с т у п н о с т и. При этом

$\theta(\nu,æ,\mu,\lambda) = 1$, если работа вида æ, проводимая с ν-м КА, может выполняться с использованием средства λ, размещенном на μ-м ОКИК;

$\theta(\nu,æ,\mu,\lambda) = 0$, в противном случае.

Так как множества A,D,B,C - конечны, то функция $\theta(\nu,æ,\mu,\lambda)$ может задаваться матрицей $\Theta = \|\theta_{\nuæ\mu\lambda}\|$, где $\theta_{\nuæ\mu\lambda} = \theta(\nu,æ,\mu,\lambda)$.

Возможности проведения различных операций управления можно характеризовать множеством X

$$X = \{<\nu, æ, \mu, \lambda, t> \in A \times D \times B \times C \times T \mid \chi_{\nu æ \mu \lambda}(t) = \varepsilon_{\nu\mu}(t)\, \theta_{\nu æ \mu \lambda} = 1\}. \qquad (12.8)$$

Возможность здесь понимается как совокупность фактов наличия зоны видимости КА-ОКИК и технической оснащенности ОКИК средствами, необходимыми для проведения данной операции.

Задача планирования операций управления заключается в выборе подмножества $X^* \subseteq X$, такого, что весь комплекс работ выполняется наилучшим в смысле выбранного критерия оптимальности образом.

Для решения данной задачи введем индикаторную функцию вида

$$u: X \to \{0, 1\}, \qquad (12.9)$$

которая принимает значение 1 или 0 в зависимости от того, запланирован соответствующий кортеж $<\nu, æ, \mu, \lambda, t>$ из множества X к выполнению или нет; таких функций $u(x)$, $x \in X$, естественно, может быть бесконечное множество. В этом случае целесообразно искать наилучшую в некотором смысле функцию $u^*(x)$, тогда

$$X^* = \{ x \in X \mid u^*(x) = 1\}.$$

Выполнение совокупности операций управления должно удовлетворять отношениям, ограничивающим выбор (ограничениям). Так, в частности, такие ограничения должны

- быть связаны с логическими зависимостями, отражающими последовательность выполнения работ с КА;
- отражать технические условия эксплуатации средств обслуживания (невозможность одновременной работы с несколькими КА, невозможность проведения ряда работ по условиям электромагнитной совместимости и т. д.).

Тогда среди всех возможных функций $u \in U$ можно выделить подмножество функций Δ, допустимых с точки зрения ограничений, накладываемых на управление КА; $\Delta \subseteq U$.

Будем полагать, что отношения предпочтения заданы в кардинальных шкалах, т.е. являются функциями вида $f_i: U \to R^1$, $i \in G$, где R^1 - множество действительных чисел. Тогда в целом задача планирования операций управления КА в общем виде может быть представлена как

$$u^* = \arg \operatorname*{opt}_{u \in \Delta} \{ f_i(u),\ i \in G \} \qquad (12.10)$$

Формализованное описание множества допустимых вариантов решений (планов) Δ и целевых функций $f_i(u)$ целесообразно проводить в рамках конкретного класса моделей планирования: статических или динамических.

12.4.2. Статическая модель планирования операций управления КА.

Существенной особенностью статических моделей является то, что здесь анализ и сравнение различных программ взаимодействия основных элементов АСУ КА производятся на всем интервале управления T целиком в условиях "замораживания" изменения состояния операций.

Каждую такую программу взаимодействия (план) можно представить бесконечномерным вектором $u = \| u_{\nu æ \mu \lambda t} \|$, $\nu \in N$, ,$æ \in K$, $\mu \in M$, $\lambda \in L$, $t \in T$, компоненты которого принимают значения из {0,1}. В то же время реальный процесс взаимодействия КА со средствами управления, проводимый с целью выполнения операции управления (ν,æ), осуществляется непрерывно в некоторых интервалах времени $(t^{н}_{\nu æ}, t^{к}_{\nu æ}]$, $t^{н}_{\nu æ}, t^{к}_{\nu æ} \in T$, где T - множество моментов времени, в которые потенциально возможно выполнение операций управления. Можно считать, что выполнение каждой операции (ν,æ) эквивалентно выполнению некоторой совокупности действий $\{y_j, j \in J^{\nu æ}\}$, где под д е й с т в и е м понимается непрерывная работа технического средства λ, размещенного на μ-м пункте, проводимая в интервале времени $(t_н, t_к]$. Формально каждое действие описывается кортежем вида $<\nu, æ, \mu, \lambda, t_н, t_к>$, и тогда множество планов выполнения операций управления U можно задать в виде множества векторов $u = \| u_j \|$, компоненты которого u_j принимают значение 0 или 1 в зависимости от того, запланировано действие $y_j = <\nu, æ, \mu, \lambda, t_н, t_к>$, или нет. Таким образом

$$U = \{ \| u_j \|, j \in J^{\nu æ}, \nu \in N, æ \in K \} . \qquad (12.11)$$

Начало и конец временного интервала, когда может выполняться действие, задается, исходя из следующих соображений:

а) введением предположения о том, что $t_н$ и $t_к$ каждого действия определяются началом и концом временного интервала взаимодействия КА с ОКИК ($\varepsilon_{\nu\mu}(t)=1$);

б) дискретизацией интервалов времени, когда может выполняться операция управления.

Программа выполнения операций управления представляет собой с одной стороны программу реализации целевого назначения КА на заданном временном интервале, а с другой стороны программу работу средств управления, обеспечивающих эту реализацию. Так как в статических моделях процесс управления рассматривается на всем интервале управления сразу, то среди всех программ управления (планов) следует рассматривать только такие, которые позволяют достигать заданного состояния комплекса операций или выполнить все необходимые для этого действия. Это лежит в основе формализации условий выполнения операций, среди которых следует различать:

а) условия, вытекающие из анализа особенностей, связанных с технологией управления КА при реализации его целевого назначения;

б) условия, вытекающие из особенностей функционирования средств управления, ограниченности их ресурсов.

Таким образом, первый вид условий отражает целевые аспекты выполнения операций управления, а второй ресурсные аспекты.

Следует отметить, что если ограничения, определяемые условиями использования технических средств обслуживания АСУ при управлении КА, носят в целом однородный характер для КА различного целевого назначения, то ограничения, определяемые технологией управления, существенно зависят от специфики целевых задач, решаемых КА. В этой связи моделью планирования операций управления КА не может служить какая-либо целостная модель, но необходимо строить систему взаимосвязанных моделей. При этом можно различать модели планирования операций для различных типов КА и модели согласования (координации) планов КА.

Модели координации планов КА строятся на основе учета ресурсных ограничений, накладываемых комплексом средств управления. При построении таких моделей широко используются методы декомпозиции, и особенности соответствующих подходов далее будут рассмотрены в 15.5. Здесь же рассмотрим вариант математической модели планирования операций управления КА, учитывающий логические связи действий.

Пусть по результатам прогнозирования динамики относительного перемещения ν-го КА и технических средств управления АСУ, размещенных на ОКИК, произведена оценка возможностей управления КА на интервале времени Т. Это позволяет сформировать множество возможных действий $Y^{\nu} = \{ y_j, j \in J^{\nu}\}$, причем каждое действие y_j определяется кортежем вида $<\nu,æ,\mu,\lambda,t_н,t_к>$. Каждому действию можно сопоставить в соответствие булевую переменную u_j, которая может принимать значение 1 или 0 в зависимости от того, запланировано j-е действие, или нет. Тогда множество возможных планов выполнения операций управления ν-го КА определяется множеством векторов с компонентами $u_j \in \{0,1\}$, $j \in J^\nu$

$$U^\nu = \{ \left\| u_j \right\|, j \in J^\nu \} . \qquad (12.12)$$

На множестве U^ν необходимо задать множество допустимых планов $\Delta^\nu \subseteq U^\nu$, отражающих технологию управления КА, которая характеризуется логической взаимосвязью запланированных действий.

Рассмотрим некоторые варианты формализованного описания логической связи действий (технологических ограничений) в виде линейных ограничений.

1. Пусть необходимо выполнить не более (не менее, или ровно) $b \in \{1, 2, 3,...\}$ действий из заданного набора с номерами из множества J. Такое условие формально можно записать в виде:

$$\sum_{j \in J} u_j \le (\ge) (=) b . \qquad (12.13)$$

2. Действия с номерами из J совместно выполняться не могут:

$$\sum_{j \in J} u_j \le 1. \qquad (12.14)$$

3. Если выполнено i-е действие, то необходимо выполнить ровно одно действие с номером из J, в противном случае действия с номерами из J выполняться не могут:

$$u_i - \sum_{j \in J} u_j = 0 . \qquad (12.15)$$

4. Если выполнено действие с номером из S, то необходимо выполнить хотя бы одно действие из Q:

$$\sum_{j \in S} u_i - \sum_{j \in Q} u_j = 0 . \qquad (12.16)$$

Рассмотренные варианты формализации технологических ограничений позволяют при достаточной изобретательности описывать разнообразные комбинаторные зависимости между действиями (или операциями), с помощью линейных ограничений, накладываемых на булевые переменные. Таким образом, в виде линейных неравенств можно представить довольно широкий класс ситуаций, возникающих в процессе управления КА, и тем самым сформировать множество допустимых планов Δ^ν.

Каждому j-му действию можно поставить в соответствие некоторую характеристику $f(u_j) = f_j$, например: качество выполнения операции, длительность работы технических средств, надежность выполнения операции и другие. Тогда можно ставить задачу нахождения оптимального в смысле заданной характеристики плана в виде

$$\sum_{j \in J^{\nu}} f_j u_j \to \underset{u \in \Delta^{\nu}}{\mathrm{Opt}} . \tag{12.17}$$

Такая задача относится к классу задач линейного булевого программирования и может быть решена на основе методов, рассмотренных в главе 5.

12.4.3. Динамическая модель планирования операций управления КА.

В случае. когда для нахождения плана выполнения операций управления КА используется динамическая модель, процесс управления рассматривается, как разворачивающийся во времени. В этой ситуации чрезвычайно важным является определение понятия состояния операции $D^{\nu}_{æ}$ (работы вида æ, проводимой с ν-м КА). Далее под с о с т о я н и е м операции управления $D^{\nu}_{æ}$ в некоторый момент времени $t \in (t_o, t_f]$ будем понимать степень ее выполнения в указанный момент времени и характеризовать эту степень величиной $x_{\nu æ}(t)$. Состояние отдельной операции может меняться только в те моменты, когда функция $\chi_{\nu æ \mu\lambda}(t) = \varepsilon_{\nu\mu}(t)\, \theta_{\nu æ \mu\lambda}$ принимает значение равное 1 (см. (12.8)). Тогда дифференциальные уравнения, описывающие изменение состояния операции управления $D^{\nu}_{æ}$, могут иметь [39] вид

$$\dot{x}_{\nu æ} = \sum_{\mu \in M} \varepsilon_{\nu\mu}(t) \sum_{\lambda \in L^{\mu}} \theta_{\nu æ \mu\lambda}\, d_{\nu æ \mu\lambda}(t)\, u_{\nu æ \mu\lambda}(t) , \tag{12.18}$$

здесь

$d_{\nu æ \mu\lambda}(t)$ - заданная функция, характеризующая интенсивность выполнения операции $D^{\nu}_{æ}$ техническим средством C^{μ}_{λ};

$u_{\nu æ \mu\lambda}(t)$ - управление - функция, принимающая значение 1 или 0, в зависимости от того, целесообразно изменение состояния операции $D^{\nu}_{æ}$ в момент времени t, или нет.

Факт выполнения операции характеризует достижение состояния $x_{\nu æ}(t)$ некоторой заранее заданной величины $a_{\nu æ}$, после этого, естественно, дальнейшее увеличение $x_{\nu æ}$ не имеет смысла. Такое условие формально может быть описано функцией $Г^1_{\nu æ}(t)$

$$Г^1_{\nu æ}(t) = \gamma_-(a_{\nu æ} - x_{\nu æ}(t)), \qquad (12.19)$$

где $\gamma_-(y)$ - функция вида: $\gamma_-(y) = \begin{cases} 1, & y > 0 \\ 0, & y \le 0 . \end{cases}$

Логическая связь различных операций управления КА может быть описана ориентированным графом без циклов, тогда каждой операции управления $D^\nu_æ$ может быть поставлено в соответствие некоторое множество $I^+_{\nu æ}$ - номеров операций, предшествующих операции $(\nu,æ)$, т.е. тех операций, выполнение которых должно предшествовать выполнению операции $(\nu,æ)$. Такое условие формально может быть описано функцией вида $Г^2_{\nu æ}(t)$

$$Г^2_{\nu æ}(t) = \prod_{i \in I^+_{\nu æ}} \gamma_+(x_{\nu i}(t) - a_{\nu i}), \qquad (12.20)$$

где $\gamma_+(y)$ - функция вида: $\gamma_+(y) = \begin{cases} 1, & y \ge 0 \\ 0, & y < 0 . \end{cases}$

Тогда в целом изменение состояния операции $(\nu,æ)$ с учетом запретов на перевыполнение операций, описываемых функцией $Г^1_{\nu æ}(t)$, и с учетом факта, что операция $(\nu,æ)$ может выполняться только после того, как будут выполнены все предшествующие операции (функция $Г^2_{\nu æ}(t)$), описывается дифференциальными уравнениями следующего вида

$$\dot{x}_{\nu æ} = \sum_{\mu \in M} \varepsilon_{\nu\mu}(t) \sum_{\lambda \in L^\mu} \theta_{\nu æ\mu\lambda}\, d_{\nu æ\mu\lambda}(t)\, u_{\nu æ\mu\lambda}(t)\, Г^1_{\nu æ}(t)\, Г^2_{\nu æ}(t), \qquad (12.21)$$

За счет выбора рациональных программ управления необходимо выполнить комплекс операций управления КА, т.е. перевести динамическую систему, описываемую дифференциальными уравнениями (12.21), из состояния $x_{\nu æ}(t_o)$ в заданное состояние $x_{\nu æ}(t_f)$

$$x_{\nu æ}(t_o) = 0,\ x_{\nu æ}(t_f) = a_{\nu æ}, \quad \nu \in N,\ æ \in K^\nu. \qquad (12.22)$$

На управляющие воздействия $u_{\nu æ\mu\lambda}(t)$ могут накладываться различного рода ограничения. Например, условие, заключающееся в том, что в любой момент времени t любая работа $D^\nu_æ$ может выполняться только одним каким-либо средством C^{μ}_{λ}, записывается в виде

$$\sum_{\mu\in M}\sum_{\lambda\in L^{\mu}} u_{\nu æ \mu\lambda}(t) \le 1,\ \ \nu\in N,\ æ\in K^{\nu}. \tag{12.23}$$

Или условие, заключающееся в том, что в том, что в любой момент времени t техническое средство C^{μ}_{λ} может выполнять только одну работу $D^{\nu}_{æ}$, записывается в виде

$$\sum_{\nu\in N}\sum_{æ\in K^{\nu}} u_{\nu æ \mu\lambda}(t) \le 1,\ \ \mu\in M,\ \lambda\in L^{\mu}. \tag{12.24}$$

Таких программ управления $u_{\nu æ\mu\lambda}(t)$, $t\in(t_o,t_f]$ может быть достаточно много, целесообразно из этого множества выбрать программу, наилучшую в смысле некоторого качества. Качество управления описывается функционалом, например, можно искать управление, позволяющее выполнить весь комплекс работ за минимальное время, тогда соответствующий функционал имеет вид

$$I_1 = \int_{t_0}^{t_f} dt \to \min . \tag{12.25}$$

Задавая функцию $f_{\nu æ\mu\lambda}(t)$, характеризующую эффективность выполнения операции $D^{\nu}_{æ}$ средством C^{μ}_{λ}, можно ставить задачу поиска управления, максимизирующего суммарную эффективность выполнения всех операций, соответствующий функционал имеет вид

$$I_2 = \sum_{\nu\in N}\sum_{\kappa\in K^{\nu}}\sum_{\mu\in M}\sum_{\lambda\in L^{\mu}} \int_{t_o}^{t_f} f_{\nu\kappa\mu\lambda}(\tau)\, u_{\nu\kappa\mu\lambda}(\tau)d\tau \to \max . \tag{12.26}$$

Таким образом, задача планирования комплекса операций управления КА, поставленная на основе использования динамических моделей, относится к классу задач оптимального управления. Качество управления описывается функционалом (12.25 или 12.26), изменение состояния динамической системы описывается дифференциальными уравнениями (12.21), ограничения на управление имеют вид (12.23), (12.24), краевые условия описываются (12.22).

Контрольные вопросы

1). В чем состоят отличительные особенности сложных систем по сравнению с простыми системами ?

2). В чем заключаются трудности принятия решений в сложных системах ?

3). Какие проблемы стоят при принятии решений в сложных системах и с использованием каких принципов они решаются ?

4). Почему АСУ КА можно называть сложной системой ?

5). В чем заключается влияние внешней среды на решения, принимаемые в АСУ КА ?

6). По каким причинам при управлении в АСУ КА необходимо учитывать множество критериев оптимизации ?

7). Каким образом влияет иерархическое строение АСУ КА на модели и алгоритмы, используемые при принятии решений ?

8). Какие основные элементы должна содержать математическая постановка задачи принятия решений при управлении КА?

9). В каких ситуациях управления в АСУ КА целесообразно использовать статические модели, а в каких динамические? Чем это определяется ?

10). Какими математическими конструкциями описывается решение в статических и динамических моделях планирования Работ, проводимых с КА ?

11). Каким образом формализуются условия выполнения операций при управлении КА с использованием статических моделей ?

12). Какие из известных методов оптимизации могут применяться при использовании статических моделей планирования работы космических средств ?

13). Каким образом интерпретируется решение в динамических моделях управления КА и какой математической конструкцией оно задается?

14) Какое существенно новое понятие, характеризующее систему, вводится в динамических моделях планирования работ, проводимых с КА?

15). Как учитываются логические условия следования операций в динамических моделях управления КА ?

16). Какие методы оптимизации применяются при использовании динамических моделей управления КА, в чем состоят особенности их использования ?

13. ПРИНЯТИЕ РЕШЕНИЙ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОГО ВОЗДЕЙСТВИЯ СРЕДЫ

13.1. Характеристика задач принятия решений в условиях неопределенности среды

Анализ сложных военно-технических систем и принятие решений при управлении ими производится в рамках системно-кибернетических исследований на основе использования формальных математических моделей. Построение таких моделей (см. 1.3) связано с выделением изучаемой системы из некоторой системы большего масштаба (метасистемы), т.е. разделения этой метасистемы на исследуемую систему и с р е д у, в условиях которой и при взаимодействии с которой функционирует военно-техническая система. Такое своеобразное разделение (декомпозиция) определяется, прежде всего, целями исследования и возможностями формализованного описания систем, при этом решение задач выбора в реальных системах всегда связано с необходимостью учета факторов н е о п р е д е л е н н о с т и, в знании среды, которая влияет на формализованное описание системы (модель) и на способы принятия решений.

В зависимости от степени имеющихся знаний о среде используется та или иная форма ее описания, при этом различают следующие варианты:

- д е т е р м и н и р о в а н н а я среда, действие которой на систему полностью определено;
- с т о х а с т и ч е с к а я среда, действие которой подчиняется известным (или неизвестным) вероятностным законам;
- ц е л е н а п р а в л е н н а я среда, действие которой подчинено определенным целям;

- н е и з в е с т н а я среда - среда, о которой в той или иной степени отсутствует информация.

Последние три формы используются для описания среды с неопределенностью.

Принятие решений в условиях с т о х а с т и ч е с к о й среды основывается на том, что объективно существующая неопределенность достаточно точно описывается вероятностными законами (функциями распределения). Последнее позволяет в ряде случаев "снять неопределенность" на основе замены случайных функций соответствующими математическими ожиданиями или вероятностями превышения случайной функцией заданного порога.

Выбор в условиях ц е л е н а п р а в л е н н о г о воздействия среды связан с ситуациями, в которых сталкиваются интересы нескольких оперирующих сторон, преследующих различные цели (как правило, известные). На основе формального описания таких ситуаций, которое включает состав оперирующих сторон, задание множеств разнообразных контролируемых ими параметров, формулирование правил, по которым производится выбор параметров и оценивается эффективность действий сторон, вырабатываются рекомендации по рациональному поведению в условиях целенаправленной среды. Особое место здесь занимает понятие конфликта или конфликтной ситуации, которая может возникнуть, например, когда на один и тот же объект (процесс) воздействует группа лиц, интересы которых не совпадают между собой. В конфликтных ситуациях поиск оптимального (рационального) варианта решения осуществляется на основе методов, развитых в теории игр.

Н е и з в е с т н а я среда характеризуется индифферентным поведением по отношению к системе. Это означает, что нет оснований говорить о том, что у среды имеется цель, например, действовать максимально во вред рассматриваемой оперирующей стороне, и в тоже время нет достаточных оснований предполагать, какие значения будут принимать параметры, характеризующие состояние среды на рассматриваемом временном интервале. В ряде случаев последняя информация может быть получена на основе наблюдения того, как изменяется среда во времени, и прогнозирования изменений этих состояний для принятия решения. Можно различать следующие пути организации процесса принятия решения в случае неизвестной среды: а) ввод гипотез о направленности воздействия внешней среды на военно-техническую систему и, соответственно, на основе взаимодействия с лицом, принимающим решение, введение аксиом поведения (аксиом пессимизма, оптимизма, минимума риска и др.); б) использование экспертных знаний о возможных состояниях среды и формализация их в той или иной форме в моделях принятия решений.

Следует отметить, что в реальных ситуациях принятия решений в сложных военно-технических системах необходимо проводить комплексное использование отмеченных форм описания неопределенности. Так, в частности, может возникнуть необходимость использования такого описания, как - целенаправленная среда с элементами стохастики или неизвестности и др.

Проблема принятия решений в условиях неопределенности последние десятилетия привлекает внимание многих ученых, как в нашей стране, так и за рубежом. Ю.Б.Гермейер, Р.Беллман, Л.Заде - наиболее известные ученые, которые внесли новые, нетрадиционные идеи в теорию и практику принятия решений. Рассматриваемая проблема достаточно сложна и имеет многочисленные аспекты исследования, этим вопросам посвящена обширная литература.

Обобщая используемые в практике управления военно-техническими системами способы, можно выделить ряд направлений совершенствования методов принятия решений в условиях неопределенности.

Для снятия неопределенности в моделях принятия решений широко используются д е к о м п о з и ц и я и а г р е г и р о в а н и е, которые являются одними из основных принципов исследования в системном анализе. При этом строится система вложенных моделей различной степени детальности (уровня агрегации), соответствующих различным временным интервалам и аспектам исследования сложных систем, в частности, выделяется несколько взаимосвязанных управленческих задач. Прежде всего это задачи п е р с п е к т и в н о г о планирования (ПП), в которых рассматриваются перспективы развития военных систем на длительных временных интервалах времени. Задача д о л г о с р о ч н о г о планирования - ДП базируется на результатах перспективного планирования и состоит в нахождении конкретного состава средств и ресурсов, которые могут использоваться военно-технической системой. При решении задач о п е р а т и в н о г о планирования определяются конкретные варианты действий, связанных с управлением космическими средствами на сравнительно небольших временных интервалах. На рис.13.1 представлена взаимосвязь указанных задач

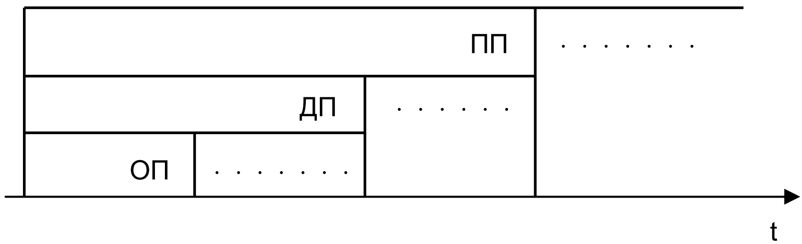


Рис. 13.1

Развиваются многошаговые методы принятия решений, при использовании которых снятие неопределенности обеспечивается за счет сокращения интервала управления и, следовательно, сокращения времени поступления возмущений. Действительно, использование предварительно найденного решения (плана управления системой) на длительных временных интервалах приводит к тому, что принятое заранее решение вследствие накапливающихся возмущений среды становится менее эффективным и не вполне соответствует реальной действительности.

В соответствии с технологией управления принятие решения (на этапе планирования) осуществляется на основе информации, полученной на момент времени t_a, с другой стороны реализуется это решение на интервале управления $(t_o,t_f]$. Тогда при управлении системой в моменты $t \in (t_o,t_f]$ необходимо учитывать возмущения, воздействующие на нее на интервале времени $(t_a,t]$. Сокращая интервал времени $(t_a,t_o]$ и интервал $(t_o,t_f]$, можно сократить количество возмущений, воздействующих на оперативный план. Другой путь повышения устойчивости управления заключается в компенсации возмущений за счет дополнения задач планирования задачами коррекции (К) планов. Тогда схема технологического цикла управления будет иметь вид

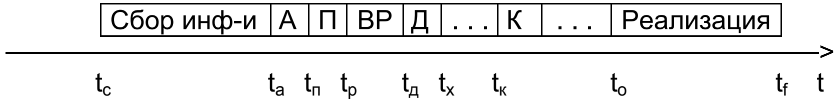


Рис. 13.2

Здесь $t_к$ - время начала решения задач коррекции плана по результатам анализа информации, полученной на интервале времени $(t_a, t_к]$.

В условиях неопределенности воздействия внешней среды математическая структура принятия решений (12.2) может быть записана в виде

$$(\Delta\times\Omega,\ f)),$$

или

$$(\Delta(\omega),\ f),\quad \omega\in\Omega, \tag{13.1}$$

здесь $\Delta(\omega)$ - множество допустимых альтернатив; $\Delta(\omega) \subseteq \Delta \times \Omega$, где Δ - собственно множество альтернатив; Ω - множество состояний среды.

f - целевая функция, характеризующая качество решения в условиях воздействия среды $f : \Delta \times \Omega \to R^1$, тогда $f = f(x,\omega)$, $x\in\Delta$, $\omega\in\Omega$.

По существу принятие решения заключается в выборе альтернативы $x \in \Delta$ в предположениях о том, какое состояние $\omega\in\Omega$ может принять среда.

13.2. Принятие решений в условиях стохастической среды

В ситуациях, когда состояния среды в массовых проявлениях повторяются с определенной частотой (такое свойство среды часто называют свойством статистической устойчивости), говорят, что среда является с т о х а с т и ч е с к о й. Частоту появления тех или иных состояний среды в этом случае можно описать количественно (вероятностью состояния).

Постановка задач принятия решений в условиях стохастической среды имеет вид

$$(\Delta(\omega),\ f(\omega)),\quad \omega\in\Omega,$$

где $\Delta(\omega)$ - множество допустимых альтернатив, $f(\omega)$ - целевая функция.

На Ω вводится вероятностная структура (W,Σ,P) по Колмогорову А.Н.:

W - рассматривается, как множество элементарных событий;

Σ - σ-алгебра случайных событий (семейство подмножеств из W, таких, что: 1) $W \in \Sigma$; 2) $\varnothing \in \Sigma$; 3) если A_i, $A_j \in \Sigma$, то $A_i \cup A_j \in \Sigma$ и $A_i \cap A_j \in \Sigma$);

P - вероятностная мера на Σ - функция, заданная на Σ и сопоставляющая каждому событию из Σ количественную характеристику (вероятность появления); P удовлетворяет аксиомам: 1) неотрицательности, $P(A) \geq 0$, $\forall A \in \Sigma$; 2) полноты, $P(W)=1$; 3) счетной аддитивности, $P(\cup A_i) = \Sigma P(A_i)$, где A_i, $i \in I$ - непересекающиеся элементы из Σ.

Частным случаем задания вероятностного пространства является случай, когда множество Ω конечно, $|\Omega| = n$. Тогда вероятностное пространство задается множеством пар $\{(\omega_1,p_1), (\omega_2,p_2), ..., (\omega_n,p_n)\}$, где $\omega_i \in \Omega$ - i-е состояние среды, p_i - вероятность i-го состояния.

Методы решения задач выбора в условиях стохастической среды можно разделить на две большие группы: методы детерминизации и методы имитационной оптимизации.

Методы детерминизации (непрямые методы) основаны на построении детерминированных эквивалентов задачи стохастического выбора. Исходной информацией для такого построения являются известные законы распределения случайных величин (состояний среды).

Методы имитационной оптимизации (прямые методы) основаны на имитации случайных изменений среды в соответствии с известными законами распределения. Для фиксированного набора случайных параметров, характеризующих состояние среды, решается оптимизационная задача. Результаты решения таких задач в последствии подвергаются статистической обработке для принятия окончательного решения.

Рассмотрим подробнее указанные методы.

13.2.1. Методы детерминизации.

При решении конкретных задач выбора на вероятностных структурах часто вводится предположение о том, что задание целевой функции $f(\omega)$ и ограничивающих отношений $r_i(\omega)$, $i=1,...,m$, определяющих множество допустимых альтернатив $\Delta(\omega)$, может быть осуществлено с помощью некоторых функций $g_i(x,\omega)$, $i = 0,1,...,m$, где каждое g_i: $\Delta \times \Omega \rightarrow R^1$, принимающие соответственно при $x \in \Delta$, $\omega \in \Omega$ значения $g_i(x,\omega)$ (причем $g_o(x,\omega) = f(x,\omega)$).

Такая задача выбора является задачей стохастического математического программирования, и ее можно представить в виде.

$$f(x,\omega) \to \max,$$

$$g_i(x,\omega) \le 0, \quad i = 1,...,n.$$

Далее будем иметь в виду обозначение $g_o(x,\omega) = f(x,\omega)$.

Функции $g_i(x,\omega)$, $i = 0,...,n$ представляют собой случайные функции, и при каждом фиксированном x являются случайными величинами с заданным законом распределения. Сущность методов детерминизации заключается в переходе от моделей с указанными случайными функциями к моделям, зависящим лишь от числовых характеристик, соответствующих законов распределения (например, математического ожидания, дисперсии и т.д.), которые являются уже детерминированными величинами.

Как правило, вводятся следующие числовые характеристики законов распределения случайной функции $g_i(x,\omega)$, $i=0,...,n$.

1). Математическое ожидание (M - преобразование)

$$g^m(x) = M\,(g(x,\,\omega)). \tag{13.2}$$

2). Дисперсия (V - преобразование)

$$g^v(x) = M(\,g(x,\,\omega) - M\,g(x,\omega))^2. \tag{13.3}$$

3). Вероятность превышения значений случайной функции некоторого заданного порогового значения b (P - преобразование)

$$g^p(x) = P\{\,g(x,\,\omega) \ge b\,\}. \tag{13.4}$$

4). Максимальное значение b, которое превышается значениями случайной функции с заданной вероятностью q (B - преобразование)

$$g^b(x) = \max\,\{b \mid P\{g(x,\,\omega) \ge b\} \ge q\,\}. \tag{13.5}$$

Ограничения задачи стохастического программирования подвергаются, как правило, однотипным преобразованиям. Поэтому в целом такие задачи характеризуются парой символов <G,C>, где первый символ характеризует вид преобразования целевой функции, а второй характеризует вид преобразований ограничений. Например, различают <M,M> задачи, в которых целевая функция и ограничения подвергаются M-преобразованию; <M,P> задачи, в которых ограничения подвержены P-преобразованию и т.д. Наиболее широко в задачах стохастического программирования используются преобразования типа M, P, B, как более полно отвечающие существу решаемых задач.

Довольно часто при поиске наилучших альтернатив функции $g_i(x, \omega)$, $i=0,...,n$ представляются в виде полиномов. Тогда эти функции можно задать как $g_i(x,h_i(\omega))$, $i=0,...,n$, где $h_i(\omega)$ - вектор коэффициентов полинома, т.е.

$$g_i(x, h(\omega)) = \sum_{j=0}^{k} h_{ij}(\omega)\, x^j .$$

В результате проведения детерминизации задач стохастического программирования их окончательное решение может быть получено уже с использованием известных методов математического программирования (см. раздел 2).

13.2.2. Методы имитационной оптимизации.

В методах имитационной оптимизации (прямых методах стохастического выбора) не производится преобразование задачи к ее детерминированному эквиваленту. Суть данных методов заключается в том, что генерируются случайные значения ω в соответствии с известным законом распределения, далее для заданного ω вычисляются функции $g_i(x, \omega)$, $i = 0,...,n$, и решается оптимизационная задача вида $x = \arg\max f(x, \omega)$, где $x \in \{x \mid g_i(x,\omega) \le 0\}$, $i=1,...,n$ при фиксированном ω. Особенностью реализации данных методов является то, что, во-первых, при произвольном выборе ω допустимое множество альтернатив может оказаться пустым (ограничения несовместны), и, во-вторых, общий объем таких вычислений для получения устойчивых результатов может быть достаточно велик.

Преодоление указанных трудностей достигается на основе применения методов и м и т а ц и о н н о г о у с р е д н е н и я и методов с т о х а с т и ч е с к о й а п п р о к с и м а ц и и.

При имитационном усреднении произвольно (случайно) выбирается некоторое подмножество точек из допустимой области при некотором значении параметров среды. Далее на каждом шаге используется некоторая процедура решения задачи нелинейного программирования и производится усреднение значений функций, ограничивающих выбор, и значений градиентов, вычисленных на множестве избранных точек.

Метод стохастической аппроксимации, как и метод имитационного усреднения, представляет собой итерационную процедуру градиентного типа. При реализации данного метода на каждом шаге процесса происходит движение к оптимуму в некотором случайном направлении в пространстве альтернатив. При этом сам итерационный процесс сходится при определенных условиях. Использование классических методов стохастической аппроксимации для решения задач выбора в стохастической среде требует их соответствующей модификации для задач условной оптимизации.

13.2.3. Модель двухэтапного стохастического выбора.

Как уже отмечалось в 13.1, реализация плана (программы управления) производится в условиях изменяющейся внешней среды. Для компенсации возмущений среды используются многошаговые методы принятия решений и, в частности, принятие решений по схеме ... $\rightarrow$ (сбор информации, анализ, прогноз) $\rightarrow$ (принятие решения) $\rightarrow$ (коррекция решения) $\rightarrow$ (реализация плана)...

Коррекция решения производится либо на этапе непосредственно предшествующем реализации плана, либо на этапе реализации плана по результатам наблюдения и оценки состояния среды. Причем под коррекцией понимают, как правило, компенсацию возмущений среды за счет заранее выделенных ресурсов (иначе такую коррекцию называют коррекцией по программе управления - плану, в отличие от коррекции по конечному состоянию).

В этой связи основным содержанием задачи двухэтапного стохастического выбора является поиск такого плана (нахождение такого решения), который с учетом последующей его коррекции позволил бы минимизировать затраты на эту коррекцию. Тогда содержание первого этапа разработки программы управления состоит в поиске указанного плана, а содержание второго этапа состоит в коррекции плана по результатам наблюдений за состоянием среды.

Формальная постановка задачи двухэтапного стохастического линейного программирования имеет вид

$$c^{T}x - M(\, d^{T}(\omega)\, y\,) \rightarrow \max, \qquad (13.6)$$

$$A(\omega)\, x + B(\omega)\, y = b(\omega)\, ,\ \omega \in \Omega, \qquad (13.7)$$

$$x,\ y \geq 0. \qquad (13.8)$$

Здесь

x - решение, принимаемое на первом этапе до наблюдения ω;

y - коррекция решения x по результатам наблюдения за состоянием среды;

ω - наблюдаемое состояние среды, которое на этапе выбора x неизвестно;

c - коэффициенты целевой функции исходной задачи выбора, которые позволяют оценить качество принятого решения;

$d(\omega)$ - вектор стоимости затрат на проведение коррекции в том случае, если среда приняла состояние ω;

$A(\omega)$ - матрица условий, ограничивающих выбор x, соответствующих состоянию среды ω;

$B(\omega)$ - матрица условий проведения коррекции;

$b(\omega)$ - реализация вектора ограничений в том случае, если среда находится в состоянии ω.

$M(d^T(\omega)\ y)$ представляет собой среднее значение (математическое ожидание) затрат на проведение коррекции в условиях ω.

Величина $(b(\omega) - A(\omega)x)$ характеризует величину возмущений (невязки), обусловленных состоянием среды ω, которые необходимо компенсировать в результате проведения коррекции.

В общем случае решение задачи при известном законе распределения ω является достаточно трудоемким, т.к. производится на основе использования методов имитационной стохастической оптимизации.

Рассмотрим случай, когда множество Ω конечно. Тогда вероятностное пространство задается множеством пар (ω_1,p_1), $(\omega_2, p_2),\ldots,(\omega_n,p_n)$, где $\omega_i\in\Omega$ - i-е состояние среды, p_i - вероятность i-го состояния. В этом случае число решений (число возможных коррекций) второго этапа конечно и равно n, соответствующее множество имеет вид $\{y_1,y_2,\ldots,y_n\}$. Решение первого этапа (план) и множество решений второго этапа (множество возможных коррекций) находятся в результате решения поставленной задачи (13.6)-(13.8), которая примет вид:

$$c^T x - \sum_{i=1}^{n} (p_i\, d_i\, y_i) \ \rightarrow\ \max, \qquad (13.9)$$

$$A_i\, x + B_i\, y_i = b_i,\ \ i = 1,\ldots,n\ , \qquad (13.10)$$

$$x,\ y_i \geq 0. \qquad i = 1,\ldots,n. \qquad (13.11)$$

Здесь $A_i=A(\omega_i)$, $B_i=B(\omega_i)$, $d_i=d(\omega_i)$, $b_i=b(\omega_i)$, $i = 1,...,n$. Матрица условий задачи (13.9)-(13.11) имеет вид

$$\begin{Vmatrix} A_1 & B_1 & 0 & \ldots\ldots & 0 \\ A_2 & 0 & B_2 & \ldots\ldots & 0 \\ . & . & . & . & . \\ A_n & 0 & 0 & \ldots\ldots & B_n \end{Vmatrix}$$

т.е. она имеет блочно-диагональную структуру с левым окаймлением. Для решения таких задач могут использоваться декомпозиционные алгоритмы решения задач большой размерности.

13.3. Принятие решений в условиях целенаправленной среды

Принятие решений в условиях целенаправленной среды связано с тем, что известна цель среды, в соответствии с которой она выбирает свои состояния и которую преследует в своих действиях. Эти действия могут вступать в противоречия с нашими действиями, т.е. формируется конфликтная ситуация.

Конфликтной ситуацией назовем ситуацию, в которой сталкиваются интересы нескольких оперирующих сторон, преследующих различные цели.

Конфликтная ситуация характеризуется: составом оперирующих сторон; целями, которые преследует каждая из сторон; способами участия оперирующих сторон в конфликте; возможными результатами разрешения (исходами) конфликта.

Формализованную модель некоторой конфликтной ситуации называют *игрой*.

Методы принятия рациональных решений в условиях целенаправленного воздействия среды интенсивно развиваются в настоящее время в рамках *теории игр*, которая входит в качестве раздела в общую теорию выбора и принятия решений. Теория игр является математической теорией моделей принятия решений при целенаправленном воздействии среды. Эта теория создает формальную основу для анализа конфликтных и противоречивых ситуаций и, в конечном итоге, позволяет формулировать рекомендации о наилучшем поведении в таких ситуациях. Ее предназначение - выработка рекомендаций по рациональному поведению участников конфликта (оперирующих сторон - игроков). Формальное описание игры предполагает задание множества участников конфликта, задание множеств разнообразных контролируемых ими параметров и формулирование правил, по которым производится выбор параметров и оценивается эффективность всевозможных действий участников. Цели участников конфликта не обязательно должны быть антагонистическими.

Конкретное описание конфликтной ситуации осуществляется путем задания определенных п р а в и л игры. В частности, в качестве таких правил могут выступать:

- возможные действия игроков;
- состав информации о действиях других игроков и об условиях, в которых происходит игра;
- оценки качества действий каждого из игроков в ходе конфликта.

Игры с различными правилами можно классифицировать по следующим признакам.

1. Количество участников игры. По этому признаку различают игры двух игроков и игры n лиц.
2. Мощность множеств выбора игроков (количество возможных вариантов действий каждого участника). Различают игры конечные и бесконечные.
3. Суммарный выигрыш игроков: игры с нулевой (ненулевой) суммой.
4. Количество розыгрышей игры. Различают одношаговые, многошаговые и дифференциальные (непрерывные) игры. Если игра многошаговая, то различают фиксацию очередности ходов - игры с фиксированной (нефиксированной) последовательностью ходов.
5. Информированность игроков - состав имеющейся у участников игры информации о действиях противника, функциях выигрыша, числе ходов и т.д. Различают игры с полной информацией и игры с неполной информацией.
6. Математико-психологические аспекты игры:

- угрозы - информированность противника о возможных последствиях его хода;
- блеф - ложная информация, сообщаемая противнику;
- рефлексия - постановка себя на место противника.

В зависимости от значений указанных признаков рассматривают различные конкретные игры. Так, например, а н т а г о н и с т и ч е с к о й и г р о й называется игра n лиц (игроков) с нулевой суммой.

Игры n лиц в зависимости от характера взаимоотношений игроков могут быть бескоалиционными и коалиционными. В первых играх между игроками не допускается никаких соглашений для принятия совместных решений, во-вторых - игроки могут составлять коалиции для принятия согласованных решений с целью увеличения выигрыша каждого игрока.

Партия игры представляет из себя фиксированный вариант реализации игры при неизменных правилах и складывается из отдельных ходов - решений, принимаемых противоположными сторонами.

Поведение каждой из оперирующих сторон (игроков) характеризуется стратегией.

Стратегией игрока называется совокупность правил, определяющих выбор варианта действий (принятия решения) при каждом личном ходе игрока в зависимости от ситуации, сложившейся в процессе игры.

В ряде случаев в связи с необходимостью отражения степени информированности игроков происходит расширение понятия стратегии. Стратегия рассматривается как функция со значениями множества выборов, вид которой и область определения зависят от имеющейся и ожидаемой информации о действиях других игроков. Различных стратегий (правил поведения) у игрока может быть достаточно много.

Оптимальной стратегией игрока принято называть такую стратегию, которая при многократном повторении партий игры обеспечивает ему максимально возможный выигрыш (или соответственно минимально возможный проигрыш).

Таким образом, основной целью изучения игр является выявление оптимальной стратегии, приводящей к максимальному выигрышу игрока. При выборе оптимальной стратегии содержание основной гипотезы о поведении противника состоит в предположении, что он разумен и делает все для увеличения своего выигрыша.

13.3.1. Постановка задач игрового выбора.

Рассмотрим формализованное представление задачи принятия решений в условиях целенаправленной среды. Обобщенную задачу принятия решения в условиях неопределенности можно записать в виде

$$(\Delta \times \Omega, \{f_j, j \in G\}),$$

где в качестве среды Ω выступают противники, преследующие свои цели f_j, $j \in \{2,\ldots,n\}$.

Т.к. имеется n оперирующих сторон, то множество альтернатив $\Delta \times \Omega$ представляется в виде декартового произведения (расчленяется)

$$\Delta \times \Omega = \Delta^1 \times \Delta^2 \times \cdot \cdot \cdot \times \Delta^n ,$$

где Δ^j - множество альтернатив j-ой оперирующей стороны (игрока), j=1, 2,..., n (здесь $\Delta^1=\Delta$). Соответственно множество целевых функций $f_j, j \in G$, заданных на $\Delta\times\Omega$ представляет собой множество функций выигрыша игроков, $f_j : \Delta^1\times\Delta^2 \times\cdots\times \Delta^n \to R^1$, $j\in G=\{1,2,...,n\}$.

Неопределенность выбора заключается в том, что каждая оперирующая сторона осуществляет выбор альтернативы $x_j\in\Delta^j$, в тоже время качество принятого решения (выигрыш) определяется функцией $f_j(x_1,..,x_n)$, то есть зависит от выбора других игроков, который заранее неизвестен.

Таким образом, каждый j-ый игрок решает задачу

$$x^*_j = \arg \max_{x_j\in\Delta^j} f_j(x_1, x_2,..., x_j,..., x_n), \tag{13.12}$$

и целью теории игр является разрешение этой неопределенной ситуации.

13.3.2. Матричные игры. Чистые и смешанные стратегии.

Простейшим вариантом игры является антагонистическая игра, в которой противодействуют две оперирующих стороны (2 игрока), при этом множества различных альтернатив, из которых они выбирают свои решения, конечны. Такая игра называется матричной.

М а т р и ч н о й и г р о й называется конечная игра двух лиц с нулевой суммой.

Тогда такая игра характеризуется конструкцией

$$(\Delta^1 \times \Delta^2, \{ f_1, f_2 \}). \tag{13.13}$$

Участники игры могут выбирать свои решения из конечного множества альтернатив

$$\Delta^1 = X = \{ x_i, i = 1,..., m \},$$

$$\Delta^2 = Y = \{ y_j, j = 1,..., n \}.$$

Т.к. игра антагонистическая, то $f_1+f_2=0$. Откуда $f_1= f$; $f_2= -f$. В такой игре может быть $m\times n$ различных исходов, которые характеризуются различными значениями выигрыша первого игрока $f_{ij} = f(x_i,y_j)$. Все эти исходы можно характеризовать прямоугольной матрицей выигрышей F (платежная матрица), элементы f_{ij} которой характеризуют выигрыш первого игрока и, соответственно, проигрыш второго при выборе игроками альтернатив, номера которых соответствуют номеру строки i и столбца j этой матрицы.

Итак, совокупность пар (x_i,y_j) характеризует множество исходов игры, а матрица, содержащая элементы соответствующие значениям выигрыша, называется п л а т е ж н о й м а т р и ц е й и имеет следующий вид

$$F = \begin{array}{c} \\ x_1 \\ x_2 \\ \\ \\ x_m \end{array} \begin{array}{c} \begin{array}{cccc} y_1 & y_2 & \dots\dots\dots & y_n \end{array} \\ \left\| \begin{array}{cccc} f_{11} & f_{12} & \dots\dots\dots & f_{1n} \\ f_{21} & f_{22} & \dots\dots\dots & f_{2n} \\ \dots & \dots & \dots\dots\dots & \dots \\ \dots & \dots & \dots\dots\dots & \dots \\ f_{m1} & f_{m2} & \dots\dots\dots & f_{mn} \end{array} \right\| \end{array}$$

В матричных играх, под стратегией понимается ход, приводящий к наибольшему выигрышу. Множество таких ходов у первого игрока равно X и его часто называют множеством ч и с т ы х с т р а т е г и й первого игрока.

Проанализируем рациональное поведение одного из игроков, например, первого. На каждую i-ю стратегию (ход, максимизирующий выигрыш) первого игрока, второй игрок ответит j-ой стратегией, которая минимизирует выигрыш первого игрока от применения его стратегии. Тогда гарантированный выигрыш первого игрока от применения i-ой чистой стратегии при использовании всех возможных стратегий второго игрока будет равен

$$F_i^{(-)}= \min f_{ij},\ j=1,\dots,n$$

В этих условиях для первого игрока наилучшим (оптимальным) поведением представляется выбор стратегии, максимизирующей значение гарантированного выигрыша $F_i^{(-)}$, соответственно наилучшей стратегией является стратегия, на которой достигается

$$F^{(-)} = \max_i\ F_i^{(-)} = \max_i \min_j f_{ij}. \qquad (13.14)$$

Это для первого игрока гарантированный выигрыш (при рациональном его поведении меньше этого значения он выиграть не может), который называется *нижней ценой игры*.

Стратегии, доставляющие первому игроку выигрыш, равный нижней цене игры называются *гарантирующими*.

Второй игрок, поступая аналогично, путем выбора своих гарантирующих стратегий может свести свой проигрыш к гарантированному минимуму

$$F^{(+)} = \min_{j} \max_{i} f_{ij} , \qquad (13.15)$$

который называется *верхней ценой игры*. Для первого игрока $F^{(+)}$ представляет собой максимально возможный выигрыш при рациональном поведении второго игрока. Обычно принято оба вида гарантирующих стратегий для краткости называть минимаксными.

Итак, первый игрок имеет m чистых стратегий, а второй игрок имеет n чистых стратегий. В каждой строке платежной матрицы F, характеризующей возможные исходы игры, первый игрок выбирает элемент матрицы, имеющий наименьшее значение, далее среди этих значений ищется наибольшее, которое и соответствует нижней цене игры. В каждом столбце матрицы второй игрок выбирает наибольшее значение, а затем среди этих значений находится наименьшее значение (наименьший проигрыш), что соответствует верхней цене игры.

Рациональным поведением игрока в такой ситуации является, очевидно, выбор гарантирующей стратегии, тогда его выигрыш будет находиться в диапазоне от $F^{(-)}$ до $F^{(+)}$.

<u>Пример.</u>

$$F = \begin{array}{c} \left\| \begin{array}{ccccc} 3 & 6 & 7 & 3 & 3 \\ 1 & 4 & 4 & 2 & 4 \\ 9 & 3 & 1 & 6 & 8 \\ 4 & 5 & 3 & 4 & 8 \end{array} \right\| \begin{array}{c} 3^* \\ 1 \\ 1 \\ 3^* \end{array} \\ \begin{array}{ccccc} 9 & 6^* & 7 & 6^* & 8 \end{array} \end{array}$$

гарантирующие стратегии 1-го игрока

гарантирующие стратегии 2-го игрока

$F^{(-)} = 3$; $F^{(+)} = 6$.

Если верхняя цена игры равна нижней $F^{(-)} = F^{(+)} = C$, то игру называют игрой, *разрешимой в чистых стратегиях*, а значения верхней и нижней цены игры называют просто **ц е н о й игры**

$$\max_i \min_j f_{ij} = \min_j \max_i f_{ij} = C. \qquad (13.16)$$

Соответствующие гарантирующие стратегии в этой ситуации являются **о п т и м а л ь н ы м и** и целесообразно выбирать именно такие стратегии.

Так как данная ситуация соответствует наличию седловой точки у платежной функции f(x,y) (функции двух векторных аргументов), то и игру называют **и г р о й с с е д л о в о й т о ч к о й.** Причем седловая точка определяется парой оптимальных чистых стратегий. Характерной особенностью такой игры является устойчивость поведения игроков, которая заключается в том, что любое отклонение одного из участников игры от своей оптимальной стратегии может привести только к большому проигрышу (в лучшем случае выигрыш останется без изменений). Так, например, в игре с платежной матрицей

$$F = \begin{Vmatrix} 0 & 1 & 4 & 1 \\ 5 & 0 & 1 & 3 \\ 4 & 2 & 3 & 4 \end{Vmatrix}$$

имеется седловая точка (x_3,y_2). Совокупность соответствующих стратегий является решением игры, при этом цена игры равна 2.

Соответствующие чистые стратегии являются устойчивыми, так как отклонение каждого из игроков от оптимального решения приводит к худшим для этого игрока результатам.

Следует отметить, что имеется большое число ситуаций, когда матричная игра не разрешима в чистых стратегиях. Так в рассмотренном ранее примере верхняя цена игры равна 6, а нижняя цена игры равна 3. Таким образом, седловая точка отсутствует. Тогда чистые стратегии становятся неустойчивыми, и выбор оптимальной чистой стратегии становится неопределенным.

Итак, в ситуациях, когда игра не разрешима в чистых стратегиях, выбор оптимальной стратегии становится неопределенным. Очевидно, что в одношаговой игре целесообразно выбирать гарантирующую стратегию. В то же время в многошаговых играх создается возможность повысить выигрыш за счет рациональной комбинации чистых стратегий, в том смысле, что бы выбирать их на каждом шаге с определенной вероятностью или частотой.

С м е ш а н н о й с т р а т е г и е й первого игрока называется упорядоченная последовательность m чисел $p = (p_1, p_2, \ldots, p_m)^T$, удовлетворяющих условиям

$$p_i \geq 0,\ i=1,\ldots,m; \qquad \sum_{i=1}^{m} p_i = 1. \tag{13.17}$$

При этом числа p_i, $i=1,\ldots,m$ интерпретируются как вероятности выбора i-ой чистой стратегии на каждом шаге игры, или как относительная частота выбора этой стратегии на нескольких шагах.

Аналогично смешанной стратегии первого игрока вводится смешанная стратегия второго игрока $q = (q_1, q_2, \ldots, q_n)^T$.

Следует отметить, что смешанная стратегия позволяет описывать и чистые стратегии. Действительно, в случае, когда $p_i=0$, $i = 1,\ldots,m$, $i \neq j$, а $p_j = 1$, смешанная стратегия соответствует чистой стратегии с номером j.

Смешанных стратегий у каждого игрока может быть достаточно много (бесконечно много), в этой связи целесообразно ставить вопрос о поиске оптимальной смешанной стратегии. Однако для того, чтобы производить поиск такой стратегии следует убедиться, существует ли она - оптимальная смешанная стратегия. Ответ на этот вопрос дает основная теорема теории игр.

О с н о в н а я т е о р е м а т е о р и и и г р (теорема о минимаксе). Любая матричная игра разрешима в смешанных стратегиях.

Таким образом, у первого игрока (соответственно и у второго игрока) с у щ е с т в у е т оптимальная смешанная стратегия, применение которой позволяет максимизировать средний выигрыш в многошаговой матричной игре.

$$\max_p \min_q p^T F q = \min_q \max_p p^T F q = C. \tag{13.18}$$

Поясним приведенную основную теорему теории игр. Запишем матрицу игры в виде $F = \{ f_{ij} \}$, где $i = 1,\ldots,m$, $j = 1,\ldots,n$. Смешанные стратегии первого игрока обозначим через p, смешанные стратегии второго игрока - через q. Тогда на каждом шаге 1-ый игрок выбирает свою i-ю чистую стратегию с вероятностью p_i, а 2-ой игрок выбирает свою j-ю чистую стратегию с вероятностью q_j.

В этом случае среднее значение величины выигрыша (математическое ожидание) определяется как

$$C = \sum_{i=1}^{m} \sum_{j=1}^{n} p_i f_{ij} q_j = p^T F q. \qquad (13.19)$$

Средний выигрыш является скалярной функцией двух векторных аргументов p и q. Тогда существует седловая точка функции среднего выигрыша.

Итак, оптимальная смешанная стратегия существует. Необходимо решить вопрос о том, как ее найти.

Пусть оптимальная смешанная стратегия p известна. Анализ оптимальной смешанной стратегии, показывает, что все ее компоненты можно разделить на две группы: равные нулю и большие нуля. Чистые стратегии, которым в оптимальной смешанной стратегии соответствуют нулевые компоненты называются *пассивными* (неактивными) чистыми стратегиями. Действительно, если оптимальный стиль поведения в многошаговой игре, который описывается оптимальной смешанной стратегией, предписывает на каждом шаге выбирать некоторые стратегии с вероятностью 0, то такие стратегии естественно назвать пассивными.

Чистые стратегии, которым в оптимальной смешанной стратегии соответствуют компоненты большие нуля, называются *активными* стратегиями.

Компоненты смешанной стратегии ассоциируются с вероятностями применения соответствующих чистых стратегий, тогда активность стратегии соответствует целесообразности ее применения в многошаговой игре. Соответственно пассивные чистые стратегии можно не рассматривать вообще, что создает основу для упрощения игры. Для выделения пассивных стратегий вводится понятие доминирующих (доминируемых) стратегий.

Чистая стратегия первого игрока с номером i *доминирует* чистую стратегию с номером k, если для элементов платежной матрицы выполняется условие $f_{ij} \ge f_{kj}$, j = 1,...,n, и хотя бы одно неравенство строгое. Тогда говорят, что i-ая стратегия, доминирует стратегию k-ую, или k-ая стратегия, доминируется стратегией i-ой. Аналогично для второго игрока чистая стратегия с номером j доминирует чистую стратегию с номером k, если выполняется условие $f_{ij} \le f_{ik}$, i = 1,...,m.

Говорят, что i-я чистая стратегия первого игрока доминируется *линейной комбинацией* некоторого подмножества L других его чистых стратегий, если выполняется условие

$$f_{ij} \le \sum_{k \in L} \lambda_k f_{kj}\ ,\ j = 1,\dots,n,\ L \subseteq \{1,\dots, m\}$$

и хотя бы одно неравенство строгое. Здесь λ_k - коэффициенты, удовлетворяющие условиям

$$\lambda_k \ge 0,\ k \in L; \quad \sum_{k \in L} \lambda_k = 1.$$

Доминируемые стратегии всегда пассивны, и наоборот, если стратегия пассивна, то она доминируется какой-либо чистой стратегией, или линейной комбинацией стратегий.

Например, пусть задана платежная матрица в виде

$$F = \begin{Vmatrix} 2 & 5 & 6 & 2 \\ 8 & 2 & 1 & 5 \\ 3 & 4 & 2 & 3 \end{Vmatrix}$$

Анализ матрицы F показывает, что 3-я стратегия первого игрока доминируется линейной комбинацией 1-ой и 2-ой стратегий с коэффициентами 2/3 и 1/3; 1-я стратегия второго игрока доминируется его 4-ой стратегией, а 2-я стратегия доминируется линейной комбинацией 3-ей и 4-ой стратегий с коэффициентами 0.75 и 0.25. Тогда игровая модель с платежной матрицей F эквивалентна модели с матрицей

$$\begin{Vmatrix} 6 & 2 \\ 1 & 5 \end{Vmatrix}$$

П р и м е ч а н и е. Сравнивая выражение, записанное при определении смешанной стратегии, и выражение для линейной комбинации стратегий, заметим, что элементы смешанной стратегии, по существу, представляют собой коэффициенты линейной комбинации всех чистых стратегий соответствующего игрока.

Существует следующее утверждение, имеющее важное прикладное значение.

Утверждение (об активных стратегиях). Выигрыш от использования оптимальной смешанной стратегии, примененной против любой активной стратегии или линейной комбинации активных стратегий, равен цене игры .

Следовательно, если игрок в процессе игры не выходит из класса своих активных стратегий, то при любом поведении противника (даже, если тот применяет оптимальную смешанную стратегию) он не проиграет больше цены игры (может проиграть меньше). В том случае, если он использует пассивные стратегии, то проигрыш может быть больше цены игры.

Тогда, если p^* - оптимальная смешанная стратегия первого игрока, то выигрыш первого игрока при использовании p^* против некоторой j-ой чистой стратегии второго игрока вычисляется как

$$p^{*T} F\, e^j \geq C\,, \qquad (13.20)$$

где e_j - вектор, такой что его компоненты $e^j_i=0$, $i=1,...,n$, $i \neq j$, а $e^j_j=1$.

Утверждение (о числе активных стратегий). В любой матричной игре с платежной матрицей F размерности ($m \times n$) количество активных стратегий каждого игрока не превосходит min {m, n} (наименьшего из чисел m и n).

13.3.3. Методы нахождения оптимальных смешанных стратегий.

Процедура нахождения оптимальных чистых или смешанных стратегий соответствует выявлению рациональной линии поведения противников в конфликтной ситуации, описываемой игровой моделью. Поэтому такую процедуру часто называют р е ш е н и е м и г р ы (интерпретируя ее как процесс), а пару оптимальных стратегий (чистых или смешанных) называют решением игры, рассматривая ее как результат.

В качестве п о д г о т о в и т е л ь н о г о этапа при решении игры рассматривается этап упрощения игры, в ходе которого выполняется:

- преобразование платежной матрицы к более простому виду;
- выделение доминируемых стратегий игроков.

В результате преобразования платежная матрица сводится к виду, когда все ее элементы неотрицательны. При проведении данного преобразования используются два свойства:

1). Если ко всем элементам платежной матрицы добавить некоторое число P, то решение игры не изменится, но цена игры увеличится на P.

2). Если все элементы платежной матрицы умножить на некоторое число Q, то решение игры останется тем же, но цена игры увеличится в Q раз.

Тогда, например, игра с платежной матрицей

$$F = \begin{Vmatrix} 0.5 & 0.6 & 0.9 \\ 0.7 & 0.5 & 0.5 \\ 0.9 & 0.7 & 0.8 \end{Vmatrix} \text{ эквивалентна игре с платежной матрицей } F^1 = \begin{Vmatrix} 0 & 1 & 4 \\ 2 & 0 & 0 \\ 4 & 2 & 3 \end{Vmatrix}$$

При этом все элементы F были домножены на 10 и из элементов полученной матрицы вычли 5, соответственно изменилась и цена игры. Решение игры осталось прежним - седловая точка (3,2); т.е. оптимальны 3-я стратегия первого игрока и 2-я стратегия второго игрока. Если цену игры, равную 2, подвергнуть обратным преобразованиям (добавить 5 и разделить на 10), то получим исходное значение цены игры 0.7.

Решение матричной игры с использованием методов линейного программирования.

Пусть задана игра с платежной матрицей $F = \{ f_{ij} \}$, в которой у первого игрока m чистых стратегий, а у второго игрока n чистых стратегий. Требуется найти решение игры, т.е. определить оптимальные смешанные стратегии игроков.

Приведем игру к виду, когда в платежной матрице содержатся только положительные элементы. Предположим, что известна оптимальная смешанная стратегия 1-го игрока $p=(p_1,p_2,\ldots,p_m)^т$. Тогда, используя утверждение об активных стратегиях (13.20), можно записать n неравенств вида

$$p^т F e^j \geq C, \quad j = 1,\ldots,n.$$

Это эквивалентно $p^т F \geq C$, или $F^т p \geq C$.

Последние неравенства можно записать в виде

$$\begin{aligned}
&f_{11}p_1 + f_{21}p_2 + \ldots + f_{m1}p_m \geq C,\\
&f_{12}p_1 + f_{22}p_2 + \ldots + f_{m2}p_m \geq C,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&f_{1n}p_1 + f_{2n}p_2 + \ldots + f_{mn}p_m \geq C.
\end{aligned}$$

Т.е. оптимальная смешанная стратегия, примененная против любой стратегии противника, дает выигрыш, не меньший цены игры. Если стратегия противника активная - выигрыш равен цене игры, если пассивная, то выигрыш может быть больше цены игры.

Разделим правую и левую часть неравенств на C (т.к. элементы платежной матрицы неотрицательны, то $C \geq 0$, и характер неравенств сохранится); введем обозначения $x_i = p_i / C$, $i=1,\ldots,m$. Тогда

$$f_{11}x_1 + f_{21}x_2 + \ldots + f_{m1}x_m \geq 1,$$
$$f_{12}x_1 + f_{22}x_2 + \ldots + f_{m2}x_m \geq 1,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots \qquad (13.21)$$
$$f_{1n}x_1 + f_{2n}x_2 + \ldots + f_{mn}x_m \geq 1,$$
$$x_i \geq 0,\ i = 1,\ldots,m.$$

Последнее условие следует из того, что p_i и C положительны. Все эти условия задают множество возможных смешанных стратегий первого игрока. Из этого множества целесообразно выделять стратегию, доставляющую максимальный выигрыш первому игроку, или, что одно и то же, стратегию, максимизирующую цену игры. Рассмотрим для этого линейную форму

$$L(x) = x_1 + x_2 + \ldots + x_m. \qquad (13.22)$$

Подставим значения x_i ($x_i = p_i/C$) и получим,

$$L(x) = 1/C\ (\ p_1 + p_2 + \ldots + p_m\ ) = 1/C.$$

Действительно, сумма p_i, $i=1,\ldots,m$ равна 1 (из определения смешанной стратегии).

Следовательно, минимизируя L(x) на множестве допустимых x, которое задается (13.21), можно найти оптимальную смешанную стратегию p, доставляющую максимум цене игры. В силу линейности целевой функции и ограничений такая задача может решаться методами линейного программирования. Решение этой задачи всегда существует, т.к. значения F неотрицательны и ищется минимум суммы неизвестных переменных.

Оптимальная смешанная стратегия определяется как $p_i = x_i/L$, $i = 1,\ldots,m$; цена игры $C = 1/L$.

Оптимальная смешанная стратегия второго игрока находится в результате решения задачи, двойственной к поставленной задаче. Тогда использование метода обратной матрицы (см. 3.5), позволяющего одновременно получать решения задач двойственной пары, обеспечивает нахождение оптимальных смешанных стратегий обоих игроков.

Метод итераций (Брауна - Роббинсон).

Метод итераций является приближенным методом решения игровых задач. При большой размерности решаемых задач и в условиях, когда не требуется высокой точности получаемых решений, данный метод достаточно эффективен.

Метод представляет собой своеобразную имитацию игры. Многократно проигрываются партии игры, при этом партнеры последовательно производят свои ходы в условиях, когда им известен предыдущий ход противника. Итак,

1) игра многократно проигрывается;

2) на каждой итерации игроками выбирается наилучшая в сложившихся условиях чистая стратегия;

3) при оценке условий анализируется суммарный, накопленный за все предыдущие итерации выигрыш;

4) при выборе стратегии игрок предполагает, что будущее будет похоже на прошлое.

В результате проведения ряда итераций относительные частоты применения чистых стратегий приближенно можно принять в качестве оптимальных смешанных стратегий игроков, а средний выигрыш является приближенным значением цены игры.

Итак, пусть 1-ый игрок выбирает некоторую стратегию (например, гарантирующую, на которой достигается нижняя цена игры), в этих условиях 2-ой игрок отвечает ему такой своей стратегией, при которой выигрыш 1-го минимален. Тогда, в условиях сделанного 2-м игроком хода, 1-ый игрок выбирает такую чистую стратегию, при которой его суммарный выигрыш за все итерации максимален. Аналогичным образом поступает 2-ой игрок с той разницей, что он стремится уменьшить суммарный за все проведенные итерации выигрыш 1-го игрока (свой проигрыш).

Пусть матричная игра задается платежной матрицей $F = \{f_{ij}\}$ размерности $(m \times n)$. Результаты всех итераций записываются в таблицу вида

Таблица

к	i	Z_{11}	......	Z_{1n}	j	Z_{21}	......	Z_{2m}	C_1	C_2	C

Здесь

к - номер итерации (номер розыгрыша партии игры);

i - номер чистой стратегии 1-го игрока, выбранной им на к-ой итерации;

Z_{1r}- суммарный выигрыш 1-го игрока при применении r-ой чистой стратегии 2-м игроком на к итерациях при различных стратегиях (r) 2-го игрока; $r = 1,...,n$;

j - номер чистой стратегии 2-го игрока,выбранной им на к-ой итерации;

Z_{2s}- суммарный проигрыш 2-го игрока при применении s-ой чистой стратегии 1-м игроком на к итерациях при различных стратегиях (s) 1-го игрока; $s = 1,...,m$;

C_1 - средний выигрыш 1-го игрока на к итерациях (его суммарный выигрыш деленный на к);

C_2 - средний проигрыш 2-го игрока на к итерациях;

$C = (C_1 + C_2) / 2$; представляет собой среднюю цену игры.

В результате проведения (к) итераций компоненты оптимальной смешанной стратегии определяются как относительные частоты появления соответствующих чистых стратегий. Иначе говоря, оценивается количество появлений в столбцах i и j таблицы чистых стратегий, выбираемых первым и вторым игроком, и полученные числа делятся на (к).

Особенность итерационного метода заключается в том, что, несмотря на довольно медленную сходимость метода, объем вычислений с ростом размерности задачи возрастает не так сильно, как это имеет место в задачах линейного программирования. Кроме того, объем данных, необходимых для запоминания при реализации метода, растет линейно с ростом размерности задачи. В этой связи итерационный метод достаточно легко реализуется на цифровых вычислительных машинах.

На основании изложенного можно сформулировать последовательность действий, которые целесообразно проводить для применения положений теории игр при анализе конфликтных ситуаций.

1. На первом этапе перечисляются и изучаются существенные возможности (альтернативы) по принятию решения, имеющиеся у каждой из оперирующих сторон.

2. Оцениваются последствия выбора альтернатив оперирующих сторон. Решение этого вопроса зависит от содержания задачи и представляет известные трудности в определении численных значений такой оценки. Для этого целесообразно использовать методы экспертного анализа.

3. Составляется платежная матрица игры (матрица эффективности выбора).

4. Оценивается возможность существования седловой точки игры. Если седловая точка существует, то выбор оптимальной стратегии определен. Если седловой точки нет, то можно утверждать, что значение цены игры лежит между верхней и нижней ценой игры, а игра разрешима в смешанных стратегиях.

5. Производится упрощение платежной матрицы на основе исключения доминируемых стратегий и приведение ее к стандартному виду (в матрице отсутствуют отрицательные элементы).

6. На основе использования одного из методов решения игры находятся смешанные стратегии оперирующих сторон и цена игры.

7. Заключительный шаг состоит в интерпретации полученных численных результатов в терминах исходной постановки задачи и выдаче практических рекомендаций по их применению при принятии решения.

13.4. Принятие решений в условиях неизвестной среды

В случае неизвестной среды нет достаточных оснований для предположений о том, какие значения будут принимать параметры, характеризующие состояние среды на рассматриваемом временном интервале. При этом возможно два направления создания информационной базы для принятия решения.

П е р в о е направление связано с наблюдением за изменением состояний среды, и затем на основе собранной информации (статистики) строится вероятностное распределение состояний среды. После этого возникает возможность использования методов выбора в условиях стохастической среды. Здесь можно различать две ситуации: а) имеются достаточные объективные предпосылки для априорного задания вида закона вероятностного распределения состояний среды и необходимо только определить (на основе сбора и статистической обработки информации) параметры этого закона (параметрические статистики); б) в ходе наблюдения за изменением состояний среды необходимо выявить сам закон вероятностного распределения состояний среды (непараметрические статистики).

При анализе возможностей принятия решения в рамках данного направления следует учитывать следующее:

1) для сбора и обработки статистики необходим ресурс времени, который на практике довольно часто отсутствует;

2) изменение состояний среды должно обладать свойством статистической устойчивости, т.е. состояния повторяются с определенной частотой в массовых явлениях. Вместе с тем в сложных системах, особенно, когда в контуре управления присутствуют люди, такое свойство довольно часто не выполняется.

В т о р о е направление связано с получением экспертной информации о целесообразном поведении в сложившейся ситуации или об оценках возможных состояний среды, которая учитывается в том или ином виде.

Рассмотрим здесь второе направление, связанное с анализом способов учета экспертной информации в моделях принятия решений.

13.4.1. Модели типа "игра с природой".

Специфическим видом игр, имеющих важное прикладное значение для анализа ситуаций, возникающих при принятии решений в неизвестной среде, являются так называемые "игры с природой". В этих играх в качестве второго игрока выступает "природа", которая не заинтересована в результатах игры и, следовательно, действует по своим законам, не противодействуя сознательно другой оперирующей стороне. К прикладным проблемам, использующим модели типа "игра с природой", можно отнести многие военно-технические задачи разработки систем, задачи планирования боевых действий в различных условиях. Неопределенность воздействия среды может определяться как принципиальной невозможностью ее изучения, так и ограничениями ресурсного характера, связанными с этим изучением. Под такими ограничениями, как правило, понимается время, материальные, финансовые затраты и т.п.

Множество с о с т о я н и й п р и р о д ы R (стратегий) интерпретируется как известное по своему составу множество состояний внешней среды. При этом предполагается, что в каждой конкретной ситуации принятия решения реализуется только один элемент из R, который при решении задачи выбора полагается неизвестным. Будем полагать, что множество состояний природы конечно $R = \{ R_1,...,R_s \}$.

Тогда возможные исходы игры можно характеризовать платежной матрицей F:

$$
F = \begin{array}{c} \\ x_1 \\ x_2 \\ \\ \\ x_m \end{array}
\begin{array}{cccc} R_1 & R_2 & \ldots\ldots\ldots & R_s \\ \left\| \begin{array}{cccc} f_{11} & f_{12} & \ldots\ldots\ldots & f_{1s} \\ f_{21} & f_{22} & \ldots\ldots\ldots & f_{2s} \\ \ldots & \ldots & \ldots\ldots\ldots & \ldots \\ \ldots & \ldots & \ldots\ldots\ldots & \ldots \\ f_{m1} & f_{m2} & \ldots\ldots\ldots & f_{ms} \end{array} \right\| \end{array}
$$

Рис.13.4

Существенным отличием данной ситуации от ситуаций, рассмотренных в условиях целенаправленной среды, является то, что здесь второй игрок (природа) не стремится действовать максимально во вред первому. Это создает возможность первому игроку повысить свой выигрыш по сравнению с гарантированной стратегией (оптимальной смешанной), в то же время неопределенность состояния природы, в которых она может находиться, требует введения лицом, принимающим решение, некоторых предположений, выраженных в аксиомах, принципах оптимизации и т.п., а также соответствующих этим предположениям критериев выбора поведения. Следует отметить, что такой неформальный ввод некоторого принципа, осуществляемый на основе экспертного анализа складывающейся ситуации, во многом определяет дальнейший поиск оптимального решения. Естественно, насколько полно будет проведен такой анализ, и насколько полно выбранный принцип оптимизации будет соответствовать реальности, настолько эффективным будет принятое решение.

Предположения (принципы, аксиомы), вводимые лицом, принимающим решение, позволяют свести ситуацию принятия решения в условиях неизвестной среды к ситуации принятия решения в условиях стохастической или целенаправленной среды.

Так, в частности, исходная ситуация сводится к стохастической среде, при введении предположения о равновероятном распределении состояний природы.

Принцип равновероятных состояний среды (критерий Лапласа)

Следует отметить, что знание вероятностного распределения состояний среды (такое знание можно интерпретировать, как знание смешанной стратегии q второго игрока) делает ситуацию вполне определенной - в этом случае целесообразно выбирать чистую стратегию из X, на которой достигается максимальный средний выигрыш (математическое ожидание выигрыша), найденный в соответствии с

$$x^* = \arg \max_i F(x_i), \text{ где } F(x_i) = F_i = \sum_{j=1}^{s} f_{ij}q_j, \ i=1,...,m.$$

Тогда, если состояния среды равновероятны (q_j=1/s), то следует выбирать решение

$$x^* = \arg \max_i \frac{1}{s} \sum_{j=1}^{s} f_{ij} = \arg \max_i \sum_{j=1}^{s} f_{ij} .$$

Существует ряд широко распространенных принципов оптимальности и соответствующих им критериев, позволяющих свести исходную неизвестную ситуацию к целенаправленной среде.

Принцип гарантированного результата (критерий пессимизма, критерий Вальда)

Это известный принцип максимина, который позволяет получить гарантированный результат для оперирующей стороны (1-ый игрок) независимо от того, в каком состоянии находится среда (даже если она будет преследовать строго антагонистические цели). Правило выбора (1 игрока) имеет следующий вид:

$$x^* = \arg\max_{x \in X} \min_{r \in R} f(x,r),$$

здесь $f(x,r)$ - функция выигрыша оперирующей стороны.

Данный принцип является выражением крайнего пессимизма, поскольку рекомендует ориентироваться на самые худшие условия.

Принцип максимального оптимизма

Этот принцип оптимальности, противоположный предыдущему, ориентирует оперирующую сторону на то, что среда максимально благоприятствует ее действиям. В указанной ситуации осуществляется выбор в соответствии с правилом:

$$x^* = \arg\max_{x \in X} \max_{r \in R} f(x,r),$$

Принцип пессимизма-оптимизма (критерий Гурвица)

На основе введения числа g, характеризующего степень оптимизма оперирующей стороны, и изменяющегося от 0 до 1, осуществляется линейная свертка первых двух правил выбора:

$$x^* = \arg\max_{x \in X} (g \max_{r \in R} f(x,r) + (1-g) \min_{r \in R} f(x,r)).$$

Поскольку g характеризует степень оптимизма оперирующей стороны, то, если $g = 0$ - получаем критерий пессимизма Вальда; если $g = 1$ - получаем критерий оптимизма.

Принцип минимума максимальных потерь
(критерий минимизации риска, критерий Сэвиджа)

Данный критерий гарантирует наименьшую величину максимально возможной потери выигрыша по сравнению с тем, который мог бы быть достигнут, если бы было известно состояние среды $r \in R$. Этот критерий аналогичен критерию Вальда, но сам пессимизм здесь понимается по другому. Вводится функция риска

$$z(x,r) = \max_{x \in X} f(x,r) - f(x,r),$$

которая характеризует отклонение выигрыша при некоторой стратегии x и состоянии среды r от максимального при данном состоянии природы выигрыша. Тогда критерий Сэвиджа имеет вид:

$$x^{*} = \arg \max_{x \in X} \min_{r \in R} z(x,r),$$

В некоторых случаях решения, выбранные по разным критериям, совпадают друг с другом, и тогда, естественно, целесообразно принять такое решение в качестве окончательного. Однако довольно часто этого не происходит, и эффективность решения определяется тем, насколько правильно будет произведена неформальная оценка неизвестных состояний среды. Тем не менее, всегда целесообразно провести предварительное исследование с использованием различных критериев, проанализировать причины несовпадения решений (если оно имеет место) и после этого принять окончательное решение.

13.4.2. Выбор на основе нечеткого описания среды.

Исследования процессов принятия решений при управлении сложными военно-техническими системами показывают, что обрабатываемая и анализируемая информация о состоянии указанных систем в условиях неизвестной среды представлена в большинстве случаев в виде понятий и отношений, задаваемых на естественном либо на профессионально-ориентированных языках. Одним из конструктивных способов формального описания ситуаций, связанных с неопределенностью принятия решения, является способ, основанный на н е ч е т к о м (размытом) описании основных элементов формализованного представления ситуации принятия решений.

Поскольку понятие "множество" лежит в основе построения формальных математических моделей, то введение нечеткости в определение множества и рассмотрение на этой основе "нечетких множеств" служит основой построения нечетких моделей принятия решений в условиях неизвестной среды. Нечеткие множества были введены Л.Заде в 1965 г., как средство описания не вполне различимых объектов и процессов окружающей действительности. И, если по определению Г.Кантора: множество - совокупность элементов (объектов), *хорошо* различимых нашей мыслью или интуицией, то можно определить, что

Н е ч е т к о е м н о ж е с т в о - совокупность объектов, *не достаточно хорошо* различимых нашей мыслью или интуицией.

Некоторое четкое множество D можно формально задать с использованием индикаторной функции

$$\mu_D : U \to \{0, 1\},$$

которая принимает значение 1 или 0 в зависимости от того входит элемент универсального множества U в множество D, или нет. Тогда

$$D = \{ x \in U \mid \mu_D (x) = 1\}.$$

Обобщая данные понятия на "не вполне различимые объекты", можно определить формальный способ задания нечеткого множества:

- нечеткое множество задается множеством пар вида

$$D = \{ (x, \mu_D(x)) \}, \qquad (13.23)$$

где $\mu_D(x)$ - степень принадлежности элемента x из универсального множества U к нечеткому множеству D, которая задается функцией принадлежности

$$\mu_D : U \to [0, 1]. \qquad (13.24)$$

Нечеткое множество представляет собой обобщение традиционного понятия множества, и, поскольку понятие нечеткости вводится в основу формального описания систем - в понятие множества, то создается возможность формального представления различного рода нечетких систем и процессов (формального ввода в эти системы неопределенности знаний исследователя).

Рассмотрим основные определения, связанные с нечеткими множествами.

- Н о с и т е л е м нечеткого множества D называется четкое множество $B(D) = \{ x \in U \mid \mu_D(x) > 0 \}$.
- В ы с о т о й нечеткого множества D называется число H(D), такое, что $H(D) = \sup \mu_D(x),\ x \in U$.
- Нечеткое множество D называется н о р м а л ь н ы м, если $H(D) = \sup \mu_D(x) = 1,\ x \in U$.
- Нечеткое множество D называется п у с т ы м, если $\mu_D(x) = 0,\ \forall x \in U$.

<u>Примеры.</u>

1). D - нечеткое множество "малых" чисел на R^1

$$\mu_D(x) = (1 + (x/10)^2)^{-1},$$

тогда $\mu_D(0) = 1$; $\mu_D(10) = 0.5$; $\mu_D(100) = 0.01$.

2). Задание нечеткого множества на примере нечеткого числа. Пусть " x <u>приблизительно</u> a ", тогда

$$\mu_D(x) = (1 + ((x-a)/q)^2)^{-1},$$

здесь q характеризует степень "размытости" нечеткого числа относительно a. Действительно, при $(x-a) = q$ степень принадлежности $\mu_D(x)=0.5$.

<u>Операции над нечеткими множествами.</u>

Над нечеткими множествами проводятся операции, аналогичные тем, которые проводятся над четкими множествами. В результате теоретико-множественных операций, проводимых над нечеткими множествами $A = \{(x, \mu_A(x))\}$ и $B = \{(x, \mu_B(x))\}$, получается нечеткое множество $C = \{(x, \mu_C(x))\}$. Следовательно, для задания нечеткого множества C необходимо определить, каким образом вычисляются значения $\mu_C(x)$ по значениям $\mu_A(x)$ и $\mu_B(x)$.

Рассмотрим основные теоретико-множественные операции.

Равенство: $A = B \iff \forall x \in U\ \ \mu_A(x) = \mu_B(x)$.
Подмножество: $A \subseteq B \iff \forall x \in U\ \ \mu_A(x)\ ,\ \mu_B(x)$.
Объединение: $C = A \cup B \iff \forall x \in U\ \ \mu_C(x) = \max \{\mu_A(x), \mu_B(x)\}$.
Пересечение: $C = A \cap B \iff \forall x \in U\ \ \mu_C(x) = \min \{\mu_A(x), \mu_B(x)\}$.
Дополнение : $C = C_u A \iff \forall x \in U\ \ \mu_C(x) = 1 - \mu_A(x)$.
Разность : $C = A \setminus B \iff \forall x \in U\ \ \mu_C(x) = \max \{\mu_A(x)-\mu_B(x), 0\}$.
Декартово произведение: $C = A \times B \iff \forall (x,y) \in U \times U\ \ \mu_C(x,y) = \min \{\mu_A(x), \mu_B(y)\}$.

Дополнительные операции над нечеткими множествами:
Растяжение: C = DIL (A) <==> $\forall x \in U \ \ \mu_C(x) = (\mu_A(x))^{1/2}$.
Концентрация: C = CON (A) <==> $\forall x \in U \ \ \mu_C(x) = (\mu_A(x))^2$.
Нормализация C = N(A) <==> $\forall x \in U \ \ \mu_C(x) = \mu_A(x) / \sup \mu_A(x)$.

Нечеткие отношения и отображения.

Нечеткое о т н о ш е н и е R на декартовом произведении множеств $A \times B$ - нечеткое множество элементов из $A \times B$, задаваемое функцией принадлежности $\mu_R: A \times B \to [0,1]$.

$$R = \{ ((x,y), \mu_R(x,y)) \}. \qquad (13.25)$$

Пример.

Пусть A=B=N; нечеткое отношение R: "x много больше y ", тогда:

$$\mu_R(x,y) = \begin{cases} \left(1 + \left(\dfrac{1}{x-y}\right)^2\right)^{-1}, & \text{если } x > y \\ 0, & \text{если } x \le y. \end{cases}$$

Поскольку нечеткое отношение - нечеткое множество, над ним выполняются все операции нечетких множеств, но кроме того есть дополнительная операция - к о м п о з и ц и я нечетких отношений.

Пусть имеются нечеткие отношения $A \subseteq X \times Y$ и $B \subseteq Y \times Z$, тогда к о м п о з и ц и е й $A \circ B$ называется нечеткое отношение $C \subseteq X \times Z$, такое, что

$$\mu_C(x,z) = \max_y \min \{\mu_A(x,y), \mu_B(y,z)\}.$$

В случае, когда X и Y - конечные множества, отношение $A \subseteq X \times Y$, как и в случае четкого множества, можно задавать матрицей M(A). Пусть $X = \{ x_1,..., x_i ,..., x_n \}$, $Y = \{ y_1,..., y_j ,..., y_m \}$, тогда $M(A) = \{ \mu_A (x_i, y_j) \}_{nm.}$. Матрицу отношения C, являющегося композицией отношений A и B, можно вычислить как

$$M(C) = M(A \circ B) = M(A) \times M(B),$$

причем операции ($\times$) соответствует (min), а операции (+) соответствует (max).

Важным частным случаем нечеткого отношения является нечеткое о т о б р а ж е н и е (функция). Можно различать три ситуации, когда результатом отображения является нечеткое множество.

1). <u>Нечеткое множество, индуцированное отображением.</u>
Пусть задано четкое отображение $f: X \to Y$ и нечеткое множество A на X: $A = \{ (x, \mu_A(x)) \}$. Тогда на Y будет индуцировано нечеткое множество $B = \{ (y, \mu_B(y)) \}$, причем

$$\mu_B(y) = \max_{x \in f^{-1}(y)} \mu_A(x), \qquad (13.26)$$

где $f^{-1}(y)$ - множество элементов из X, имеющих своим образом элемент $y \in Y$.

2). <u>Нечеткое отображение множества.</u>
Нечеткое отображение $f: X \to Y$ - нечеткое отношение, следовательно, оно задается функцией принадлежности вида $\mu_f(x,y)$. Тогда, если на X задано четкое множество A, то оно отображается в нечеткое множество B на Y. Причем

$$\mu_B(y) = \max_{x \in A} \mu_f(x,y). \qquad (13.27)$$

3). <u>Нечеткое отображение нечеткого множества.</u>
Пусть имеется нечеткое отображение $f: X \to Y$ с функцией принадлежности $\mu_f(x,y)$, и на X задано нечеткое множество A. Тогда $A = \{ (x, \mu_A(x)) \}$, заданное на X, отображается в $B = \{ (y, \mu_B(y)) \}$, заданное на Y. При этом $\mu_B(y)$ вычисляется по правилу:

$$\mu_B(y) = \mu_{f(A)}(y) = \max_{x \in A} \min \{ \mu_A(x), \mu_f(x,y)\}. \qquad (13.28)$$

Если A и B - конечные множества $A = \{ a_1,...,a_n\}$, $B = \{b_1,...,b_m\}$, то нечеткому множеству A можно сопоставить вектор m_A, компонентами которого являются степени принадлежности $\mu_A(x)$, а нечеткому множеству B можно сопоставить вектор m_B, компонентами которого являются степени принадлежности $\mu_B(y)$. То есть, $m_A = (\mu_A(a_1) ,\ldots, \mu_A(a_n))$, $m_B = (\mu_B(b_1),..., \mu_B(b_m))$,
тогда

$$m_B^{\mathrm{т}} = m_A^{\mathrm{т}} M_f,$$

где M_f - матрица отображения f.

Ситуация принятия решений (в том числе и в условиях неизвестной среды) характеризуется парой (Δ, f), где Δ - множество допустимых альтернатив, а f - целевая функция (отображение) вида f: $\Delta \to R^1$. Здесь Δ формируется на основе учета условий, которым должно удовлетворять допустимое решение. Такие условия описываются множеством $\{r_i, i\in C\}$ - отношений, ограничивающих выбор и отражающих основные пространственно-временные, технические и технологические ограничения. В ситуациях, когда отношения $\{r_i, i\in C\}$ строятся на нечетких множествах, они, в свою очередь, сами являются нечеткими множествами (13.25); обозначим такие множества, как D_i, $i\in C$. Тогда множество допустимых альтернатив Δ также является нечетким множеством, которое формируется как

$$\Delta = \bigcap_{i\in C} D_i \,. \tag{13.29}$$

Из множества Δ выбирается наилучшее в смысле критерия оптимальности, задаваемого с использованием f, решение x^*. Поскольку множество Δ, является нечетким множеством, то на множестве R^1 - множестве оценок решений $x\in\Delta$, также формируется нечеткое множество оценок G, которое в зависимости от способа задания f определяется в соответствии с (13.26)-(13.28). Нечеткость полученного множества оценок решений есть следствие нечеткости описания исходной задачи. Однако для принятия решений необходимо конкретно выбрать одну альтернативу. В качестве такой альтернативы целесообразно выбрать решение, соответствующее максимальной степени принадлежности на множестве G, т.е.

$$x^* = \arg \max_{x\in U} \mu_G(f(x)) \,, \tag{13.30}$$

здесь U – универсальное множество, на котором задано нечеткое множество оценок G.

Лингвистическая переменная.

Дальнейшее развитие способов принятия решений в неизвестной среде связано с использованием лингвистических переменных [33]. В рамках лингвистического подхода ситуация принятия решения, под которой понимаются условия и цели, описывается фразами, соответствующими термам из терм-множеств лингвистических переменных, введенных для формализованного описания ситуации.

Л и н г в и с т и ч е с к а я п е р е м е н н а я (L_p) описывается математической конструкцией вида

$$L_p = (I, T, U, Г, M). \qquad (13.31)$$

Рассмотрим составляющие L_p.

I - и м я лингвистической переменной - некоторое выражение, определяющее область действия лингвистической переменной. Например, "решение", "ограничение", "правило", "число", "удовлетворительный выбор",...

Т - т е р м - м н о ж е с т в о - множество значений лингвистической переменной. В терм-множестве различают п р о с т ы е и с о с т а в н ы е термы. Например, пусть I = "решение", тогда элементами T могут быть простые термы: "хорошее", "плохое",... и составные термы: "не хорошее", "очень хорошее", "достаточно хорошее", "очень хорошее и не очень плохое",...

Г - с и н т а к с и ч е с к о е п р а в и л о - правило (совокупность правил), которое служит для образования составных термов из простых. Составные термы образуются из простых термов и других составных термов на основе введения

- лингвистических неопределенностей ("очень", "вполне", "достаточно" и т.д.)
- лингвистических связок (и, или, не).

Некоторому часто используемому составному терму может придаваться вид простого терма - образованного простого терма. Например, "плохое" <=> "не хорошее". Это позволяет упростить составные термы, сделать их более легкими для восприятия. Так терму "очень хорошее и не очень не хорошее" может соответствовать терм "очень хорошее и не очень плохое".

U - у н и в е р с а л ь н о е м н о ж е с т в о - множество объектов (значений), на котором каждому терму (простому или составному) сопоставляется нечеткое множество - смысл терма (значение лингвистической переменной).

M - с е м а н т и ч е с к о е п р а в и л о - правило (совокупность правил), сопоставляющее значениям лингвистической переменной смысл - нечеткое множество в U. Таким образом, простым и составным термам сопоставляется некоторое нечеткое множество, причем

- для простого терма функция принадлежности задается непосредственно (на основе взаимодействия с ЛПР),
- для составных термов функция принадлежности вычисляется. При этом

- лингвистическим неопределенностям соответствуют некоторые функциональные преобразования функции принадлежности, например, "очень" - операция концентрации CON; "достаточно" - операция растяжения DIL;
- лингвистическим связкам соответствуют операции над нечеткими множествами: связке "или" соответствует операция объединения нечетких множеств; связке "и" - пересечение; "не" - взятие дополнения в универсальном множестве.

Связь составляющих лингвистическую переменную понятий можно характеризовать схемой на рис. 13.5.

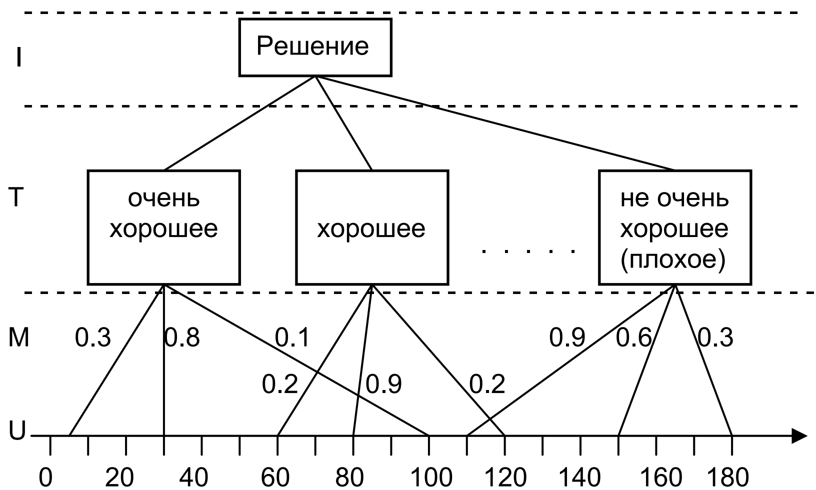


Рис. 13.5

В рамках лингвистического подхода ситуация принятия решения, под которой понимаются условия и цели, описывается фразами, соответствующими термам из терм-множеств лингвистических переменных, введенных для формализованного описания ситуации. Каждому терму, описывающему i-е условие r_i, $i \in C$ соответствует нечеткое множество D_i; терму, описывающему цель f, соответствует нечеткое множество G.

Использование лингвистических переменных позволяет ЛПР уточнять нечеткие цели и ограничения в ходе поиска оптимального решения. При этом конструируются составные термы, соответствующие новым предпочтениям ЛПР и новому пониманию им роли ограничений в формировании альтернатив. Это создает формальную базу для построения интерактивных процедур принятия решения, основывающихся на понятии лингвистической переменной.

Структура процедуры принятия решений на основе лингвистического подхода обобщенно может быть представлена как

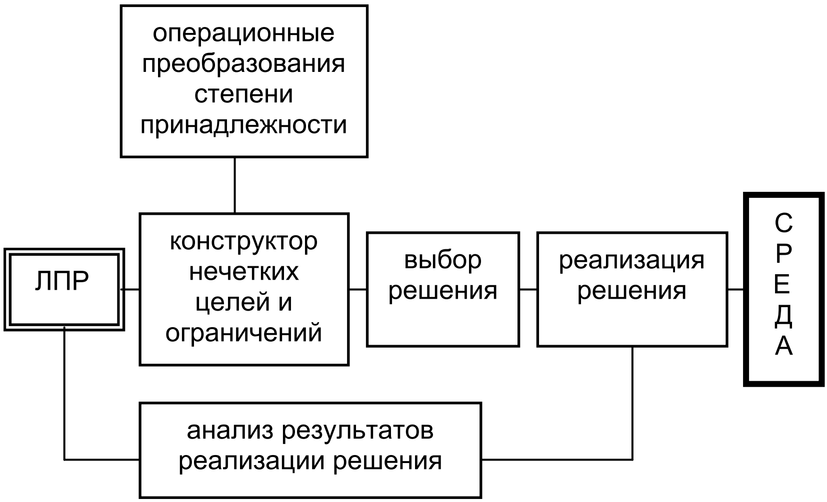


Рис.13.6

Дальнейшее развитие лингвистических моделей принятия решений основывается на введении нечетких алгоритмов, под которыми понимается упорядоченное множество нечетких инструкций, обеспечивающих при своей реализации приближенное решение заданной проблемы.

Контрольные вопросы

1. В чем состоит проблема принятия решения в условиях неопределенности воздействия среды ?
2. Что такое стохастическая среда, чем она характеризуется ?
3. Какие методы принятия решений используются в условиях стохастической среды ?
4. Как найти оптимальное решение, если число состояний стохастической среды конечно и множество допустимых решений также конечно ? В чем здесь суть детерминизации ?

5. Что такое целенаправленная среда, в чем ее особенности ?

6. Какие модели используются при принятии решения в условиях целенаправленной среды ?

7. Что такое конфликтная ситуация ?

8. Дайте классификацию игр и поясните примерами.

9. Что такое матричная игра ?

10. Определите понятие стратегии, оптимальной стратегии.

11. Какую стратегию целесообразно выбирать при однократном принятии решения в целенаправленной среде ?

12. При каких условиях гарантирующая стратегия становится оптимальной ?

13. Что такое "седловая точка", когда она существует в матричной игре и как называются стратегии, соответствующие седловой точке ?

14. В чем смысл гарантированного результата и гарантирующей стратегии для обеих оперирующих сторон ?

15. Можно ли повысить гарантированный выигрыш при многократном принятии решения, и как ?

16. Поясните смысл процесса принятия решения на основе известной оптимальной смешанной стратегии.

17. Что такое активные и пассивные стратегии, в чем их особенности и практический смысл ?

18. В чем заключаются доминирование стратегий ?

19. Какие игровые задачи целесообразно решать методами линейного программирования ? Что служит формальной основой для использования методов линейного программирования.

20. Какие задачи целесообразно решать итерационным методом. В чем его суть ?

21. Найдите результат применения (выигрыш) оптимальной смешанной стратегии первого игрока против некоторой чистой стратегии второго игрока. Сопоставьте результат применения с ценой игры, сделайте выводы.

22. Что такое неизвестная среда, чем она характеризуется ?

23. Какие критерии можно использовать при принятии решений на основе модели типа игры с природой ? Поясните их смысл и отличия ?

24. В чем суть лингвистического подхода к принятию решения в условиях неизвестности ?

25. Что такое лингвистическая структура? Поясните связь составляющих ее элементов.

14. ПРИНЯТИЕ РЕШЕНИЙ В УСЛОВИЯХ КРИТЕРИАЛЬНОЙ НЕОПРЕДЕЛЕННОСТИ

14.1. Характеристика задач многокритериального выбора

При принятии решений по управлению военно-техническими системами на первое место в современных условиях выдвигаются задачи обеспечения эффективного целевого функционирования систем с ориентацией преимущественно на их качественные характеристики. В то же время обстановка применения военных систем характеризуется частой сменой целевых установок, существенной неопределенностью функционирования. В этих условиях к принимаемому решению предъявляется, как правило, большое число разнородных требований, которым оно должно удовлетворять, и тогда качество решения необходимо оценивать с учетом этих требований, для чего используется несколько критериев оптимальности. Таким образом, достижение адекватности математического описания проблем принятия решений сложной обстановке выдвигает на передний план требование рассмотрения данных задач, как задач выбора с многими критериями (с несколькими отношениями предпочтения, на основе отношения мультипредпочтения, с векторным критерием оптимальности). Это приводит, в свою очередь, к появлению в задачах принятия решений специфического вида неопределенности - критериальной неопреде-ленности.

Суть данного вида неопределенности в задачах выбора заключается в том, что оптимальное решение, найденное по одному из критериев, может быть существенно различным по сравнению с решением, найденным по другому критерию.

Основная особенность задач многокритериального выбора заключается в том, что данные задачи не являются корректными в рамках аксиоматики, принятой в классической теории оптимизации и принятия решений. Действительно, на основе различных критериев оптимизации получаются различные решения, не совпадающие в общем случае друг с другом. Регуляризация таких задач строится на основе привлечения дополнительной качественной и количественной информации о свойствах критериальных функций, о свойствах множества альтернатив и достигается путем введения некоторых аксиом (принципов оптимальности) о том, что следует понимать под оптимальным в целом решением в этой неопределенной ситуации. Следует особо отметить, что эта дополнительная информация получается от лица, принимающего решение и несущего за него ответственность (ЛПР). Формальным выражением аксиомы (системы аксиом) является решающее правило, представляющее собой либо некоторый оператор, позволяющий сформировать результирующую целевую функцию, либо некоторый алгоритм, позволяющий выделить оптимальное решение. Естественным требованием, предъявляемым к решающим правилам, является требование выделения на их основе недоминируемых решений - решений, не улучшаемых по всем показателям одновременно.

Один из способов разрешения критериальной неопределенности, основанный на сведении исходной многокритериальной задачи к традиционной постановке задачи математического программирования или задачи оптимального управления динамическими системами, заключается, по существу, в замене всех целевых функций, кроме одной, выступающей в качестве критерия, соответствующими ограничениями. Однако такая замена, совершаемая во имя перевода задачи на "обкатанный математический путь", не получает во многих задачах соответствующего обоснования, зачастую носит волюнтаристический характер и для достаточно сложных ситуаций выбора может привести к большим просчетам. Поэтому на практике большое значение приобретает разработка новых методов принятия решений, учитывающих разнородные показатели качества решения и соответствующие им критерии.

Итак, традиционная последовательность действий при принятии решения в задачах многокритериального выбора дополняется процедурами задания аксиом оптимизации и соответствующего решающего правила, которые проводятся на основе тесного взаимодействия с лицом, принимающим решение. Выбор решающего правила обычно представляет собой существенно не-формальное действие (за исключением некоторых достаточно редких случаев, когда ввод решающего правила строго обоснован как, например, принцип минимакса в антагонистических играх), поэтому по результатам анализа найденного решения задачи многокритериального выбора может возникнуть необходимость некоторого изменения (модификации) решающего правила. Будем далее рассматривать ситуации, когда критерии задаются с использованием целевых функций. Такие задачи часто называют задачами векторной оптимизации.

Ситуацию принятия решения S (см. (1.2)) формально можно описать конструкцией следующего вида

$S = (\Delta,\ \{f_i(x),\ i \in G\}),$

где Δ - множество допустимых альтернатив; $x \in \Delta$;

$f_i(x)$ - i-я целевая функция, позволяющая оценивать качество альтернативы x;

G - множество индексов целевых функций.

Будем далее предполагать, что решается задача вида

$$x^* = \arg \underset{x \in \Delta}{\mathrm{Max}}\ f_i(x),\ i \in G\ , \qquad (14.1)$$

Можно различать два подхода к решению задач векторной оптимизации, которые различаются способом учета информации, получаемой от ЛПР. Это подходы, основанные на построении:

- операторного решающего правила;
- лингвистического решающего правила.

Операторное решающее правило.

В этом случае процесс принятия решений можно характеризовать схемой:

$$S \xrightarrow{\text{ЛПР}} A \longrightarrow F \longrightarrow f_{рез} \longrightarrow x^*, \qquad (14.2)$$

где A - аксиома (система аксиом, принцип оптимальности, схема компромисса) о том, что следует понимать под оптимальным решением;

F - решающее правило - оператор, позволяющий сформировать результирующую целевую функцию $f_{рез}$, или алгоритм, позволяющий выделить оптимальное решение x^*; $F: \{f_i(x), i \in G\} \to f_{рез}$;

x^* - оптимальное в целом решение.

Принцип оптимальности и соответствующее ему решающее правило существенно зависят от информированности ЛПР об относительной важности критериев оптимальности в задаче многокритериального выбора. Здесь можно различать три группы ситуаций, оказывающих существенное влияние на выбор конкретного решающего правила:

- отсутствие информации об относительной важности целевых функций;
- информация об относительной важности целевых функций задана в "гибкой" форме - допускается компенсация ухудшения значений более важных показателей за счет значительного улучшения менее важных;
- критерии лексикографически упорядочены ("жесткая" форма задания приоритетов критериев) - не возможна компенсация ухудшения значений более важных показателей.

Лингвистическое решающее правило.

Другим способом организации процесса принятия решений в ситуациях многокритериального выбора является формирование нечетких множеств "рациональных" решений по каждому отдельному критерию, затем на основе использования лингвистического подхода строится л и н г в и с т и ч е с к о е решающее правило, которое позволяет выделить оптимальное в целом решение. В этом случае процесс принятия решения можно характеризовать схемой:

$$S \xrightarrow{\text{ЛПР}} \{M_i,\ i \in G\} \xrightarrow{\text{ЛПР}} L \longrightarrow x^*, \tag{14.3}$$

где $M_i = \{(x,\ \mu_i(x))\}$ - нечеткое множество "хороших" (рациональных) по i-му критерию решений;

L - лингвистическое решающее правило.

Лингвистическое решающее правило L, по существу, представляет собой некоторый составной терм (см. понятие лингвистической переменной (13.31)), смысл которого задается нечетким множеством. Тогда оптимальное решение x^* соответствует максимальной степени принадлежности элемента универсального множества к нечеткому множеству, сформированному на основе анализа L.

Критериальное пространство решений. Нормализация.

В задачах векторной оптимизации, характеризуемых парой $S= (\Delta, \{f_i(x), i=1,...,s\})$, каждому решению $x \in \Delta$ можно сопоставить s значений целевых функций $f_i(x)$, $i=1,...,s$. Тогда можно рассматривать s-мерное пространство с координатами f_i значений целевых (критериальных) функций, которое в задачах векторной оптимизации часто называют, как критериальное пространство решений. Можно показать, что в задачах линейного программирования, в которых множество допустимых решений Δ описывается некоторым многогранником в n-мерном пространстве решений, образ Δ в критериальном пространстве также является многогранником.

Для применения решающих правил важно, чтобы показатели качества решений изменялись в одном масштабе. Соответствующая операция сведения диапазонов изменения целевых функций к единому масштабу называется нормализацией целевых функций. В частности, в задачах максимизации показателей довольно часто нормализацию осуществляют следующим путем

$$f^{н}(x) = \frac{f(x)}{f(x^*)}, \qquad (14.4)$$

где $f^{н}(x)$ - нормализованная целевая функция; $f^{н}(x) \in [0,1]$;
x^* - оптимальное решение.

Такой способ целесообразно использовать в случаях, если значения $f(x) \geq 0$, $\forall x \in \Delta$. Когда f(x) знакопеременна в допустимой области Δ, нормализация проводится в соответствии с

$$f^{н}(x) = \frac{f(x) - f^{l}(x)}{f(x^*) - f^{l}(x)}, \qquad (14.5)$$

где $f^{l}(x) = \operatorname{Inf} f(x),\ x \in \Delta$.

Далее будем полагать, что целевые функции нормализованы.

14.2. Множество недоминируемых решений

В задачах многокритериального выбора можно выделить некоторое подмножество множества допустимых альтернатив, элементы которого обладают определенными характерными свойствами. Это множество неулучшаемых одновременно по всем критериям (целевым функциям, показателям качества решения) альтернатив. Такое множество часто называется множеством недоминируемых альтернатив (областью компромиссов, множеством Парето, множеством эффективных решений). Дадим определение понятия недоминируемого (эффективного) решения.

Решение x^p называется **недоминируемым** (Парето-оптимальным, эффективным) решением, если в множестве Δ не существуют решения, которое лучше x^p по всем целевым функциям одновременно.

Тогда множество недоминируемых решений Δ^p (множество Парето) можно определить как

$$\Delta^p = \{\, x \in \Delta \mid \neg \exists y \in \Delta,\ f_i(y) \geq f_i(x),\ i \in G \,\}. \tag{14.6}$$

Естественно, что $\Delta^p \subseteq \Delta$, т.е., выделяя множество Парето, можно сократить область поиска оптимального решения. В то же время в задачах оптимизации множество допустимых решений Δ является, как правило, бесконечным множеством (за исключением некоторых случаев, когда Δ дискретное), тогда и множество Парето Δ^p содержит, как правило, бесконечное множество решений. В этой связи понятие множества недоминируемых решений имеет скорее методическое значение, чем практическое. Это значение состоит в том, что решающие правила (принципы оптимальности) в задачах многокритериального выбора следует строить так, чтобы решение выбиралось заведомо из множества Парето.

Рассмотрим некоторые основные свойства множества Парето, вытекающие из его определения.

Свойство 1. Никакие альтернативы, принадлежащие множеству допустимых альтернатив Δ, не доминируют (не превосходят) альтернативы, принадлежащие множеству Парето Δ^p.

Свойство 2. При переходе от одной точки (альтернативы) множества Парето к другой точке множества Парето происходит увеличение значений одних критериальных функций и уменьшение значений других критериальных функций.

Свойство 3. Множеству Парето принадлежат все альтернативы, при которых достигается экстремумы значений хотя бы одной из критериальных функций.

Важное свойство множества недоминируемых решений, которое имеет место в ситуациях, когда Δ представляет собой замкнутое и ограниченное множество, а все $f_i(x)$ непрерывны на Δ, формулируется в следующем виде.

Свойство 4. Любая точка множества Парето, расположенного в строго положительном ортанте критериального пространства, может быть найдена в результате решения задачи

$$x^p = \arg \underset{x \in \Delta}{\mathrm{Max}} \sum_{i=1}^{s} \alpha_i f_i(x), \qquad (14.7)$$

при соответствующем подборе коэффициентов α_i, на которые накладываются условия

$$\sum_{i=1}^{s} \alpha_i = 1, \quad \forall \alpha_i \geq 0.$$

Для исследования множества Парето часто используется понятие критериального пространства, под которым понимается s-мерное пространство векторов вида $f = \| f_i(x) \|$, $i=1,...,s$, $x \in \Delta$. В задачах линейного программирования при s = 2 критериальное пространство решений будет иметь вид

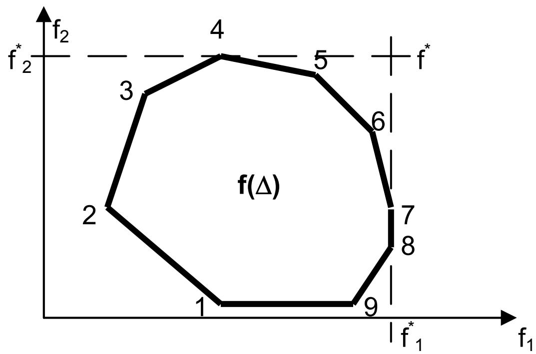


Рис.14.1.

Пусть в критериальном пространстве крайние точки многогранника допустимых решений Δ имеют образы в виде точек 1,...,9. Тогда в соответствии с определением множества Парето недоминируемыми будут решения, соответствующие граням (4,5), (5,6), (6,7) многогранника $f(\Delta)$.

Важную роль в задачах векторной оптимизации играет точка в пространстве критериальных функций, в которой все критериальные функции принимают экстремальные значения (в данном случае максимальные значения). Такая точка называется *идеальной (утопической) точкой* множества Парето. Нахождение идеальной точки f^* во многих случаях оказывается весьма полезным при постановке задач определения рациональной альтернативы. Кроме того, компоненты (координаты в пространстве критериев) утопической точки часто используются для нормализации соответствующих критериальных функций.

<u>Способы построения множества Парето.</u>

Приведенные свойства множества Парето могут быть использованы для построения и последующего исследования данного множества в конкретных прикладных задачах. Рассмотрим три таких способа.

<u>1. Используется 4-е свойство множества Парето.</u>
Тогда последовательно перебирая значения α_i, $i \in G$, в задаче

$$x^p = \arg \max_{x \in \Delta} \sum_{i \in G} \alpha_i f_i(x) , \qquad (14.8)$$

можно получать различные точки множества Парето. Вместе с тем число таких точек может быть бесконечно, поэтому свойство 4 имеет скорее методическое значение, заключающееся в том, что в ряде случаев решающие правила можно строить в форме (14.7).

<u>2. Алгоритм Джу-Зелени [15, 73].</u>

Для задач линейного программирования был разработан конечный алгоритм выделения множества недоминируемых решений. В соответствии с этим алгоритмом

- выделяется множество недоминируемых крайних точек многогранника условий задачи линейного программирования;
- формируются недоминируемые грани как линейная комбинация крайних точек;
- формируется множество Парето, как объединение недоминируемых граней.

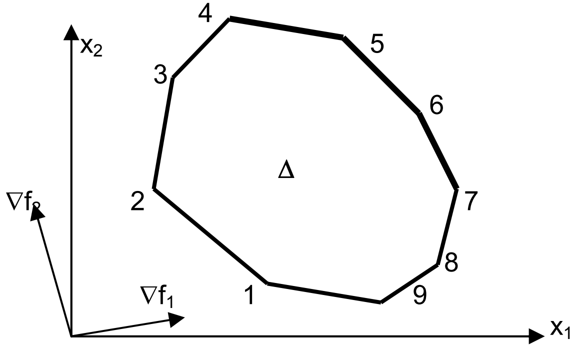


Рис.14.2.

Здесь недоминируемые крайние точки множества допустимых решений это точки 4,5,6,7, соответственно множество Парето представляет собой объединение граней (4,5), (5,6), (6,7).

3. Выделение конечного подмножества недоминируемых решений.

В задачах принятия решений довольно часто встречаются ситуации, в которых целесообразно выделять некоторое конечное множество недоминируемых решений "равномерно" распределенных в области Парето, которое выдается ЛПР для последующего неформального анализа. Выделение такого множества строится на основе вычисления диапазонов изменения значений целевых функций в области Парето, и последующего исследования допустимых решений внутри указанных диапазонов.

Рассмотрим задачу линейного программирования с векторным критерием оптимальности в следующем виде:

$f_i(x) \to \max,\ \ i=1,...,s,$

$A\,x = b,$

$x \geq 0.$

Для нахождения диапазонов изменения целевых функций решаем s оптимизационных задач вида

$$x^*_i = \arg \underset{x \in \Delta}{\mathrm{Max}}\ f_i(x),\ \ i=1,...,s\ ,$$

где $\Delta = \{\, x \in R^n \mid A\,x = b,\ x \geq 0 \,\}$.

Пусть s = 2. Тогда диапазон изменения 2-ой целевой функции в области Парето не превышает значения δf_2:

$$\delta f_2 = f^*_2 - f_2(x^*_1),\ \ \text{здесь}\ \ f^*_2 = f_2(x^*_2).$$

Зададим число N, характеризующее количество недоминируемых решений, которое следует выделить из области Парето. Множество значений 2-ой целевой функции, которые она принимает в области Парето φ_{2j}, j=1,...,N, можно задать как

$$\varphi_{2j} = f^*_2 - (j-1)\,(\delta f_2 / (N-1)\,),\ \ j = 1,....,N.$$

Теперь, выбирая первую целевую функцию в качестве ведущей (в качестве функции, по которой ищется оптимум), будем решать N задач линейного программирования вида

$f_1(x) \to \mathrm{Max}$

$f_2(x) \geq \varphi_{2j},\ \ j=1,....,N,$

$A\,x = b,\ x \geq 0.$

В результате решения N таких задач получим конечное множество x^p_j решений из области Парето.

В критериальном пространстве данный метод можно пояснить следующим рисунком.

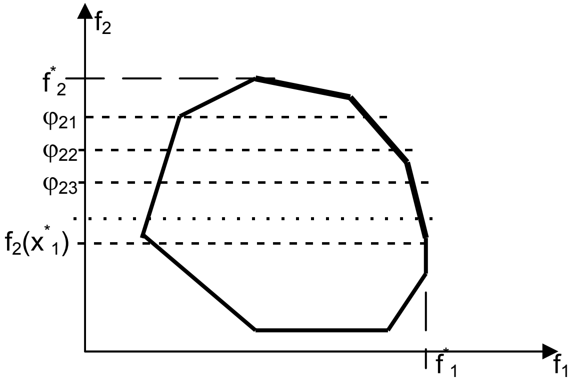


Рис.14.3.

Можно обобщить данную методику на случай s целевых функций. Выберем некоторую r-ю целевую функцию в качестве ведущей, тогда диапазон изменения i-ой целевой функции не превышает значения δf_i:

$$\delta f_i = f^*_i - f_i(x^*_r),\ i=1,...,s;\ i\neq r,$$

здесь $f^*_i = f_i(x^*_i)$, где x^*_i - решение исходной задачи;

$$x^*_i = \arg \max_{x\in\Delta} f_i(x),$$

Множество значений i-ой целевой функции φ_{ij}, $j=1,...,N_i$, которые она принимает в диапазоне изменения δf_i можно определить как

$$\varphi_{ij} = f^*_i - j\,(\delta f_i / N_i),\ i=1,...,s;\ j=1,....,N_i.$$

Теперь можно решать задачи линейного программирования вида

$$f_r(x) \to \max$$

$$f_i(x) \geq \varphi_{ij},\ i=1,...,s;\ j=1,....,N_i,\ i\neq r,$$

$$A\,x = b,$$

$$x \geq 0.$$

В результате решения N_o таких задач, где $N_o = \Sigma N_i$, i=1,.. ..., s; $i \neq r$, получим множество решений $\{ x^p_j \}$, "равномерно" распределенных в области Парето.

14.3. Принятие решений на основе операторных решающих правил

При использовании операторного решающего правила процесс принятия решений можно характеризовать схемой (14.2):

$$S \xrightarrow{\text{ЛПР}} A \longrightarrow F \longrightarrow f_{рез} \longrightarrow x^*$$

В этом случае исходная ситуация принятия решения $S = (\Delta, \{f_i(x), i=1,...,s\})$ подвергается анализу лицом, принимающим решение (ЛПР), во взаимодействии с которым формируется аксиома А (принцип оптимально\сти, схема компромисса) о том, что следует понимать под оптимальным решением в ситуации S. Формальным выражением аксиомы А является решающее правило F, который представляет собой оператор, позволяющий сформировать результирующую целевую функцию $f_{рез}$. Тогда исходная неопределенная ситуация преобразуется в ситуацию $(\Delta, f_{рез})$, в которой оптимальное решение x^* может быть найдено на основе традиционных методов оптимизации.

Используемая аксиома (принцип оптимальности) существенно зависит от информированности ЛПР об относительной важности целевых функций, которая может быть задана количественно или в форме отношений порядка. Здесь можно различать три ситуации:

1) информация об относительной важности критериев отсутствует;

2) имеется информация, заданная количественно, о возможной компенсации ухудшения значений более важных показателей за счет значительного улучшения менее важных;

3) имеется информация о важности критериев, заданная в форме отношения строгого порядка.

Рассмотрим особенности принятия решения в указанных ситуациях.

14.3.1. Решающие правила в задачах бесприоритетной оптимизации

В этом случае на основе взаимодействия с лицом, принимающим решение (ЛПР), вводится некоторая система аксиом А о том, что следует понимать под оптимальным решением. На основе системы аксиом А формируется решающее правило F, позволяющее найти в такой "доопределенной" задаче оптимальное решение x^*.

Рассмотрим конкретные примеры задания аксиом и соответствующих им решающих правил.

Аксиома 1. "Оптимальное решение соответствует максимуму минимальных значений показателей".

Такое решение может быть найдено на основе следующего решающего правила

$$x^* = \arg \underset{x \in \Delta}{\mathrm{Max}} \, \underset{i \in G}{\mathrm{Min}} \; f_i(x), \qquad (14.9)$$

где Δ - множество оцениваемых альтернатив; G - множество индексов показателей качества решения; G = {1,..., s}.

В случае, если $\Delta = \{x_1,...,x_n\}$ конечное (например, это конечное множество недоминируемых альтернатив, найдено на основе процедуры, изложенной в 14.2), можно построить матрицу F значений показателей из G при различных решениях из Δ

$$F = \begin{Vmatrix} f_{11} & f_{12} & \ldots\ldots & f_{1n} \\ f_{21} & f_{22} & \ldots\ldots & f_{2n} \\ \ldots & \ldots & \ldots\ldots & \ldots \\ \ldots & \ldots & \ldots\ldots & \ldots \\ f_{s1} & f_{s2} & \ldots\ldots & f_{sn} \end{Vmatrix}$$

где $f_{ij} = f_i(x_j)$.

Теперь, используя способы принятия решений в условиях неизвестной среды, а именно в условиях "игры с природой", можно находить гарантирующие решения (стратегии) ЛПР (в такой игре ЛПР выступает в качестве второго игрока),

$$x^* = \arg \underset{j}{\mathrm{Max}} \, \underset{i}{\mathrm{Min}} \; f_i(x_j),$$

решения, соответствующие определенной доле оптимизма g,

$$x^* = \arg \underset{j}{\mathrm{Max}} (g \, \underset{i}{\mathrm{Max}} \; f_i(x_j) + (1-g) \, \underset{i}{\mathrm{Min}} \; f_i(x_j))$$

решения, минимизирующие максимальный риск $z_i(x_j)$

$$x^* = \arg \underset{j}{\mathrm{Min}} \underset{i}{\mathrm{Max}}\, z_i(x_j).$$

Отметим, что последние два способа (решающих правила) нахождения оптимального в целом решения соответствуют самостоятельным аксиомам.

Аксиома 2. "Оптимальное решение соответствует минимуму суммы относительных отклонений от оптимальных решений по частным показателям".

Относительное отклонение $d_i(x)$ значения i-го показателя для решения x от оптимального его значения можно вычислить как

$$d_i(x) = \frac{\delta_i(x)}{f_i(x^*_i)}, \qquad (14.10)$$

где

x^*_i - решение, оптимальное по i-му показателю;

$\delta_i(x)$ - абсолютное значение отклонения i-го показателя от оптимального значения $\delta_i(x) = f_i(x^*_i) - f_i(x)$.

Тогда оптимальное в целом решение находится как

$$x^* = \arg \underset{x\in\Delta}{\mathrm{Min}} \sum_{i\in G} d_i(x), \qquad (14.11)$$

Аксиома 3. "Оптимальное решение минимизирует общую верхнюю грань относительных отклонений от оптимальных решений по частным показателям".

Пусть w - общая верхняя грань относительных отклонений от оптимальных решений по частным показателям. Тогда $d_i(x)\le w$, $i\in G$.

Оптимальное в целом решение x^* находится в результате решения задачи

$$x^* = \arg \underset{x\in\Delta}{\mathrm{Min}} \{ w \mid d_i(x)\le w,\ i\in G \}, \qquad (14.12)$$

Аксиома 4. "Оптимальное решение соответствует максимальному приближению к идеальной точке".

Идеальная точка в пространстве критериев имеет координаты $f_i(x^*_i)$, $i \in G$. Тогда квадрат расстояния до идеальной точки в евклидовой метрике определяется в виде

$$r^2(x) = \sum_{i \in G} (f_i(x^*_i) - f_i(x))^2 .$$

В этом случае оптимальное в целом решение ищется как

$$x^* = \arg \min_{x \in \Delta} r^2(x) = \arg \min_{x \in \Delta} \sum_{i \in G} (f_i(x^*_i) - f_i(x))^2, \qquad (14.13)$$

<u>Аксиома 5.</u> "Оптимальное решение соответствует "равноценному" снижению всех показателей".

Здесь в пространстве нормализованных критериев через вершины многогранника условий, соответствующие оптимальным по частным показателям решениям, проводится гиперплоскость. Направляющий вектор данной гиперплоскости соответствует результирующей целевой функцией.

Будем искать $f_{рез}$ в виде

$$f_{рез}(x) = \sum_{i \in G} \alpha_i f_i(x) .$$

Полагаем, что $f_i(x)$, $i \in G$ - нормализованные целевые функции. Тогда, в условиях того, что $f_{рез}(x)$ проходит через крайние точки многогранника, соответствующие оптимальным по частным показателям решений, α_i можно найти из системы уравнений

$$\sum_{i \in G} \alpha_i f_i(x^*_j) = c . \; j \in G.$$

Здесь $f_i(x^*_j)$ - значение i-го показателя в вершине многогранника, оптимальной по j - му критерию. Значение (c) выбирается таким образом, чтобы выполнялись условия нормировки $\alpha_i \geq 0$, $\sum \alpha_i = 1$.

В условиях отсутствия информации об относительной важности показателей используются различные решающие правила, и, соответственно, получаются различные решения. В силу наименьшей определенности такой ситуации целесообразно выявлять множество решений, соответствующих различным решающим правилам, и вычислять оценки этих решений, с тем, чтобы расширить информационную базу ЛПР для принятия решения, наиболее полно соответствующего реальной ситуации и его (ЛПР) предпочтениям.

14.3.2. Учет относительной важности критериев

Различают ситуации принятия решения, в которых существует возможность компенсации ухудшения значений более важных критериев за счет значительного улучшения менее важных. Такие ситуации часто называют ситуациями с "гибким" заданием приоритетов. Информация об относительной важности критериев может задаваться и учитываться в двух формах: 1) в виде коэффициентов относительной важности; 2) в виде компонент вектора приоритетов.

Рассмотрим особенности задания этих форм относительной важности критериев.

1). *Коэффициенты относительной важности* задаются в следующем виде.

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_s); \text{ причем } \alpha_i \geq 0, i=1,\ldots,s; \sum_{i \in G} \alpha_i = 1 . \qquad (14.14)$$

Здесь α_i характеризует относительную важность i-го критерия на множестве G. Таким образом при задании коэффициентов относительной важности необходимо оценивать относительную важность i-го показателя на множестве всех остальных, что для эксперта представляется достаточно трудоемкой процедурой. С другой стороны, задание α_i, $i \in G$, позволяет строить решающее правило в форме

$$f_{рез}(x) = \sum_{i \in G} \alpha_i f_i(x) .$$

что является формой неявного задания множества недоминируемых решений (области Парето) (см. 14.7), а, следовательно, решение, найденное на основе такой $f_{рез}(x)$, будет заведомо недоминируемым.

2). *Компоненты вектора приоритетов* упорядочены по важности и задаются как

$$\lambda = \| \lambda_1, \lambda_2, \ldots, \lambda_s \|; \text{ здесь } \lambda_j \in [1, \infty), j = 1,\ldots,s. \qquad (14.15)$$

Во второй форме задания относительной важности показателей компоненты λ_j характеризуют количество единиц j показателя, позволяющее скомпенсировать снижение (j-1)-го показателя на одну единицу (при этом λ_1=1). Такая информация может быть получена в результате экспертного опроса. С другой стороны коэффициенты относительной важности α_i по известным λ_j могут быть найдены из системы уравнений:

$$\begin{aligned}
&\sum_{i=1}^{s} \alpha_i = 1, \\
&\alpha_1 = \lambda_2 \alpha_2, \qquad\qquad (14.16)\\
&\ldots\ldots\ldots\ldots \\
&\alpha_{s-1} = \lambda_s \alpha_s,
\end{aligned}$$

или в соответствии с формулой

$$\alpha_i = \frac{\lambda^1_i}{\sum\limits_{i \in G} \lambda^1_i}, \qquad (14.17)$$

где λ^1_i коэффициент приоритета i-го показателя по отношению к 1-му показателю

$$\lambda^1_i = \left(\prod_{j=1}^{i} \lambda_i \right)^{-1}, \qquad (14.18)$$

здесь $\lambda_1 = 1$.

В этой связи целесообразно первоначально формировать ряд приоритетов (14.15), а затем с использованием формул (14.16) или (14.17) вычислять значения коэффициентов относительной важности с тем, чтобы определить $f_{рез}(x)$ и найти оптимальное решение x^* в результате решения задачи (14.7).

14.3.3. Оптимизация по последовательно применяемым критериям

Данные ситуации характеризуется наибольшей определенностью, заключающейся в том, что специфика и смысл критериев таковы, что они могут быть жестко упорядочены по важности. Тогда можно задать ряд предпочтения критериев $f = (f_1,f_2,\ldots,f_s)$, в котором каждый предшествующий критерий считается несравненно более важным, чем последующий (лексикографическое упорядочение критериев); такие ситуации называют "жестким заданием приоритетов". Вместе с тем довольно часто, исходя из способов расчета целевых функций, вычислительных погрешностей и с учетом специфики показателей, допускается некоторое отклонение от оптимальных значений показателей (можно назначить уступку).

В обоих случаях поиск оптимального решения осуществляется на основе последовательного сужения множества допустимых альтернатив:

$$x^* \in \Delta^s \subseteq \Delta^{s-1} \subseteq \ldots \subseteq \Delta^2 \subseteq \Delta^1 \subseteq \Delta. \qquad (14.19)$$

Здесь x^* - оптимальное решение; Δ^i - сужение множества допустимых альтернатив, произведенное на основе i-го показателя. Здесь

$$\begin{aligned} &\Delta^1 = \{ x \in \Delta \mid f_1(x) \geq \max_{x\in\Delta} f_1(x) - \varepsilon^1\}, \\ &\Delta^2 = \{ x \in \Delta^1 \mid f_2(x) \geq \max_{x\in\Delta^1} f_2(x) - \varepsilon^2\}, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \qquad (14.20) \\ &\Delta^s = \{ x \in \Delta^{s-1} \mid f_s(x) \geq \max_{x\in\Delta^{s-1}} f_s(x) - \varepsilon^s\}, \end{aligned}$$

где ε^i, i=1,...,s - уступка по i-му показателю. Если $\varepsilon^i = 0$, то последняя схема соответствует оптимизации по лексикографически упорядоченным критериям.

14.4. Принятие решений на основе лингвистического решающего правила

Анализ процессов принятия решений при управлении военно-техническими системами показывает, что обрабатываемая и анализируемая информация о целесообразном в конкретных условиях решении представлена в большинстве случаев в виде понятий и отношений, задаваемых на естественном либо на профессионально-ориентированных языках. Одним из конструктивных способов принятия решения в этих условиях является способ, основанный на *нечетком описании* предпочтений лица, принимающего решение. Схему принятия решения (см. (14.3)) в этом случае можно представить в виде

$$S \xrightarrow[1]{\text{ЛПР}} \{M_i,\ i \in G\} \xrightarrow[2]{\text{ЛПР}} L \xrightarrow[3]{} C(L) \xrightarrow[4]{} M_L \xrightarrow[5]{} x^*, \qquad (14.21)$$

Рассмотрим последовательность действий в соответствии с этой схемой.

(1). На основе анализа ситуации принятия решения S при взаимодействии с ЛПР вводится совокупность нечетких множеств, определяющих смысл понятия "рационального (хорошего) по i-му показателю решения". Нечеткое множество X задается конструкцией (см. 13.4.2) вида $X = \{ (x, \mu_x(x))\}$, где $x \in U$ (U - универсальное множество), а $\mu_x(x)$ - степень принадлежности элемента (x) универсального множества U к нечеткому множеству X. Степень принадлежности задается функцией вида μ_x: $U \to [0,1]$, и отражает степень уверенности ЛПР в принадлежности решения (x) к нечеткому множеству X.

В качестве универсального множества U здесь целесообразно рассматривать конечное подмножество множества недоминируемых решений Δ^p (14.6), найденное, например, на основе процедуры, изложенной в 14.2.

Нечеткие множества M_i, $i \in G$ "хороших по i-му показателю решений", задают, по существу, смысл простых первичных термов из терм множества T лингвистической переменной (13.31)

$$M_i = \{(x, \mu_i(x))\} , \quad x \in U, i \in G,$$

где U - подмножество множества недоминируемых альтернатив Δ^p.

Степень принадлежности $\mu_i(x)$ формируется на основе взаимодействия с ЛПР и может задаваться, например, как

$$\mu_i(x) = \frac{f_i(x) - f^l_i(x)}{f_i(x^*) - f^l_i(x)}, \quad i \in G, \tag{14.22}$$

или

$$\mu_i(x) = \frac{(f_i(x) - f^l_i(x))^2}{(f_i(x^*) - f^l_i(x))^2}, \quad i \in G, \tag{14.23}$$

Здесь $f^l_i(x)$ выбирается таким образом, чтобы $\mu_i(x) \in (0,1]$. Тогда

$$f^l_i(x) = \min_{x \in U} f(x) - \varepsilon. \tag{14.24}$$

Здесь ε некоторое малое число; $\varepsilon > 0$.

(2). Задание первичных простых термов (значений лингвистической переменной "решение") создает основу для формирования составных термов (в соответствии с синтаксическими правилами). Лингвистическое решающее правило L, задаваемое на основе взаимодействия с ЛПР, и представляет собой составной терм, который на профессионально-ориентированном языке определяет, что следует понимать под оптимальным в целом решении в исходной неопределенной ситуации S.

(3). Производится синтаксический разбор C(L) лингвистического решающего правила L. Т.е. составной терм представляется в виде совокупности первичных простых термов, лингвистических связок и лингвистических неопределенностей.

(4). Вычисляется нечеткое множество M_L, описывающее смысл лингвистического решающего правила. Нечеткие множества, описывающие смысл первичных простых термов, были построены в п.1. Составной терм (лингвистическое решающее правило) образуется на основе взаимодействия с ЛПР из простых и других составных на основе введения лингвистических неопределенностей (например, "очень", "вполне", "достаточно" и т.д.) и лингвистических связок ("и", "или", "не"). Синтаксический разбор, проведенный в п.3, позволяет вычислить M_L. При этом лингвистическим связкам соответствуют операции над нечеткими множествами: "или" - объединение; "и" - пересечение; "не" - взятие дополнения в универсальном множестве U, а лингвистическим неопределенностям соответствуют некоторые функциональные преобразования степени принадлежности. В частности, лингвистической неопределенности "очень" часто сопоставляется операция концентрации нечеткого множества, т.е.

$$\mu\,(\text{"очень хорошее"}) = \mu^2(\text{"хорошее"}), \tag{14.25}$$

лингвистической неопределенности "достаточно" сопоставляется операция растяжения нечеткого множества,

$$\mu(\text{"достаточно хорошее"}) = \mu^{1/2}(\text{"хорошее"}).$$

Таким образом, вычисляется нечеткое множество

$$M_L = \{\,(x,\ \mu_L(x))\},\ x \in U$$

(5). Производится выбор оптимального решения x^*, которому соответствует максимальная степень принадлежности в M_L. Т.е.

$$x^* = \arg \underset{x \in U}{\mathrm{Max}}\ \mu_L(x)\ . \tag{14.26}$$

Формируя и модифицируя лингвистические решающие правила, ЛПР может эффективно влиять на процесс выделения наиболее "рационального" в целом решения.

Контрольные вопросы

1. В чем состоит некорректность задач многокритериального выбора с точки зрения классической теории оптимизации и принятия решений ?

2. Поясните способы и пути разрешения критериальной неопределенности.

3. В чем состоит принцип Парето ?

4. Дайте определение недоминируемой по Парето (эффективной) альтернативы.

5. Перечислите и прокомментируйте основные свойства множества Парето.

6. В чем заключается основная идея выделения конечного подмножества недоминируемых решений ?

7. Какие ситуации принятия решений различаются в зависимости от информированности ЛПР. Какие ситуации являются самыми определенными? Какие являются самыми неопределенными?

8. В чем смысл проведения операции нормализации целевых функций и зачем она проводится ? Что может случиться, если такую операцию не проводить ?

9. Приведите примеры решающих правил в ситуации отсутствия информации об относительной важности критериев.

10. Какая точка критериального пространства называется идеальной ? Приведите пример.

11. В каких формах может быть учтена информация об относительной важности критериев? В чем достоинства и недостатки различных форм ?

12. На каком принципе основывается выбор решения в ситуации последовательно применяемых критериев?

13. В чем суть метода последовательных уступок? Покажите на примере задач линейного программирования, как реализуется этот метод.

14. В чем суть лингвистического подхода к принятию решений в ситуациях векторной оптимизации ?

15. Что представляет из себя лингвистическое решающее правило с точки зрения лингвистической переменной ?

16. Что представляют из себя и каким образом используются синтаксические и семантические правила лингвистической переменной в задачах векторной оптимизации ?

17. Каким образом создается необходимая информационная база для принятия решения в задачах векторной оптимизации ?

15. ПРИНЯТИЕ РЕШЕНИЙ В ИЕРАРХИЧЕСКИХ ВОЕННО-ТЕХНИЧЕСКИХ СИСТЕМАХ

15.1. Принцип декомпозиции в задачах выбора

Большинство современных военно-технических систем относится к классу, так называемых, сложных (больших) систем или систем крупного масштаба. Развитие систем управления в них связано с ростом сложности и масштабности решаемых целевых задач. Существенной особенностью такого развития является специализация функций, которая приводит к тому, что существующая система управления разбивается на совокупность подсистем, решающих специализированные задачи, которые имеют информационную, методическую и алгоритмическую общность. Это сопровождается децентрализацией процесса обработки информации и, как следствие этого, децентрализацией принятия решения. Действительно, динамичность обстановки и связанный с этим рост потоков информации может привести к тому, что полностью централизованный сбор и обработка информации либо технически невозможны, либо приводят к значительному запаздыванию в принятии решения. Указанные особенности требуют разбиения исходной системы на совокупность связанных, но самостоятельно функционирующих подсистем.

Появление в системе отдельных звеньев, способных с необходимой оперативностью перерабатывать всю поступающую информацию и принимать решения в рамках своей компетенции означает, по существу, появление в системе иерархической структуры, при которой одни подсистемы подвергаются декомпозиции, а процесс управления децентрализации. Однако такое разбиение исходной системы служит источником н о в о й н е о п р е д е л е н н о с т и. Действительно, каждая подсистема принимает решение в соответствии со своими собственными целями, не тождественными в общем случае целям других подсистем и системы в целом. В этой ситуации возникает задача с о г -

л а с о в а н и я (координации) решений подсистем, и одна из подсистем вышестоящего иерархического уровня (Центр) наделяется специальными полномочиями по решению этой задачи. По существу, координационная задача служит для учета эмерджентных свойств системы, подвергнутой декомпозиции.

Формально, с теоретико-множественных позиций декомпозиция системы представляет собой следующее.

Пусть система S задана множеством допустимых альтернатив Δ и заданными на этом множестве отношениями предпочтения F; $S = (\Delta, F)$. Произведем декомпозицию системы S на (p+1) подсистем S_i, i=0,..., p. При этом используем правила последовательной и параллельной теоретико-множественной декомпозиции систем, рассмотренные, например, в [40] .

Проведем последовательную декомпозицию системы S, тогда $S = S_o \circ S^1$, где $S_o = (\Delta_o \times P, F_o)$, $S^1 = (P \times \Delta_1 \times \cdots \times \Delta_p, F^1)$, здесь P - множество связующих сигналов; $F = F_o \circ F^1$.

Далее, используя правила параллельной декомпозиции, S^1 можно представить в виде совокупности подсистем

$S^1 = S_1 \circ S_2 \circ \cdots \circ S_p$,

где $S_1 = (P \times \Delta_1, F_1)$,

$S_2 = (P \times \Delta_2, F_2)$,

....

$S_p = (P \times \Delta_p, F_p)$,

$F^1 = F_1 \circ \cdots \circ F_p$.

Таким образом, система S в целом задается как $S = S_o \circ S_1 \circ \cdots \circ S_p$, т.е. исходная система S представлена в виде совокупности подсистем. Такое представление можно иллюстрировать схемой

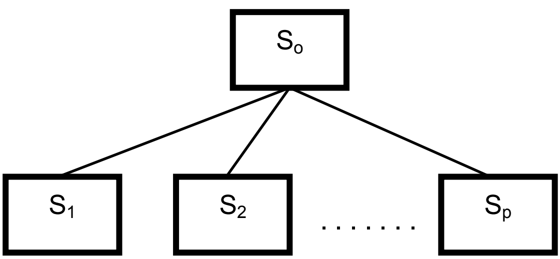


Рис.15.1.

Здесь S_o - подсистема более высокого, чем подсистемы S_j, j = 1,..., p уровня. S_o решает задачу $(\Delta_o \times P, F_o)$, подсистемы S_j, j = 1,...,p решают задачи $S_j = (P \times \Delta_j,\ F_j)$. Здесь P - множество связующих (координирующих) сигналов. Подсистему вышестоящего уровня S_o часто называют ц е н т р о м (координатором), а соответствующую ему задачу - к о о р д и н и р у ю щ е й задачей. Задачи подсистем S_j называют л о к а л ь н ы м и (задачами подсистем). Задачу принятия решений в системе S в целом называют г л о б а л ь н о й задачей. Таким образом, произведена декомпозиция глобальной задачи на координирующую и локальные задачи подсистем. Поясним особенности задачи Центра и задач подсистем, возникающих при таком разбиении.

Пусть отношение предпочтения F задается с использованием функции $f: \Delta \to R^1$, тогда задача принятия решения в системе в целом имеет вид

$$x^* = \arg \max_{x \in \Delta} f(x).$$

При этом $\Delta = \Delta_o \times \Delta_1 \times \cdots \times \Delta_p$; тогда $x = (x^o, x^1, \ldots, x^p)$; $F = F_o \circ F_1 \circ \cdot \circ F_p$; тогда $f = (f_o, f_1, \ldots, f_p)$.

Задача центра (координирующая задача) $S_o = (\Delta_o \times P,\ F_o)$ имеет вид

$$\max_{\substack{x^o \in \Delta_o \\ \pi \in P}} f_o(x^o, \pi)$$

Задачи подсистем (локальные задачи) $S_j = (P \times \Delta_j,\ F_j)$, могут различаться в зависимости от способа учета координирующих сигналов, рассчитанных в координирующей задаче. При этом различают две ситуации:

- модификация целевой функции $\max\limits_{x^j \in \Delta_j} f_j(\pi, x^j)$;
- модификация альтернатив $\max\limits_{x^j \in \Delta_j(\pi)} f_j(x^j)$.

В целом процедура выработки решения строится по итерационной схеме. На каждом шаге Центр решает свою оптимизационную задачу и вычисляет координирующий сигнал π, который сообщается подсистемам. Подсистемы на основе полученного координирующего сигнала π решают свои оптимизационные задачи и сообщают свои решения Центру, что является основой для последующего уточнения π. Процесс заканчивается, когда координирующий сигнал перестает изменяться. В результате определяются оптимальные решения подсистем $x^{*j}, j = 1,\ldots,p$ и оптимальный координирующий сигнал π^*.

Развитие исследований, связанных с проблемами декомпозиции и координации в сложных системах, началось сравнительно недавно, но прошло достаточно сложный путь, связанный с различием подходов научных школ, занимающихся этими вопросами. При этом можно сделать ряд замечаний.

В о - п е р в ы х, следует различать вопросы иерархической организации вычислительных процедур оптимизации, которые по своей форме являются процедурами выбора с мультипредпочтением, и вопросы иерархической по своему содержанию оптимизации, связанные с адекватным моделированием и оптимизацией некоторой реальной иерархической системы. Во многих случаях иерархическая организация вычислений связана с исследованием исходной иерархической системы, и тогда можно говорить об определенном соответствии формы процедур оптимизации реальному содержанию решаемой задачи. Здесь можно говорить о применении декомпозиции и координации в чисто вычислительных интересах, когда исходная оптимизационная задача большой размерности подвергается декомпозиции и разбивается на задачи существенно меньшей размерности. В этом случае координация служит для учета общих связей подсистем и позволяет достигать адекватности решений исходной задачи и задач подсистем (локальных задач), выделенных в результате декомпозиции. В ходе решения координирующей задачи вырабатываются сигналы (координирующие), которые учитываются при решении локальных задач и обеспечивают сходимость вычислительного процесса к глобальному оптимальному решению.

В о - в т о р ы х, применение координации, когда она имеет содержательное значение, т.е. когда выбор окончательных решений в координирующей задаче и в локальных задачах соответствует выбору в реальных звеньях организационно-технической структуры сложной системы. Здесь следует обратить внимание на существование двух классов задач оптимизации, связанных с реальными исходными иерархическими системами:

а) задачи выбора, в которых в постановочной части задачи для иерархической системы вводится единое отношение предпочтения, тогда целевые функции подсистем строятся на основе декомпозиции глобальной целевой функции системы и как бы "навязываются" подсистемам;

б) задачи выбора с независимо вводимыми отношениями предпочтения подсистем. Задачи последнего класса являются по своему содержанию (а не только по форме организации вычислительной процедуры) задачами выбора в иерархической структуре с мультипредпочтением. Такие задачи можно разделить на два подкласса:

1) задачи межуровневого равновесного выбора;

2) задачи, связанные с вводом координирующих воздействий, формируемых подсистемой верхнего уровня.

Межуровневый равновесный выбор имеет место в тех случаях, когда осуществляется приведение задачи выбора в двухуровневой (многоуровневой) системе к одноуровневой схеме равновесия по Нэшу. В данном случае элементы верхних и нижних уровней с точки зрения вводимых предпочтений объявляются равноправными. Такого рода решение задач выбора называется равновесным согласованием выбора в иерархической системе.

Рассмотрим задачи выбора в иерархической системе, связанные с введением координирующих сигналов.

К о о р д и н а ц и о н н ы м в ы б о р о м называется такой выбор в иерархической системе, при котором подсистема верхнего уровня воздействует на подсистемы нижних уровней посредством координирующих сигналов, влияющих на выбор этих подсистем. Для игровых постановок задач координационного выбора подсистема верхнего уровня в указанной ситуации изменяет правила игры (в настоящее время интенсивно развиваются так называемые иерархические игры). В случае оптимизационных постановок задач координационного выбора "механизмы" выбора, имеют различные интерпретации: "механизм штрафов", "механизм цен", "механизм распределения ресурсов" и др. В рамках данного направления можно различать следующие ситуации:

1) глобальная целевая функция строится на основе композиции целевых функций подсистем с учетом их относительных важностей, а центр координирует (согласует) действия подсистем на основе учета совместной операционной области их решений (ограниченность общих ресурсов, логическая зависимость решений и др.);

2) подсистемы и центр имеют собственные интересы и соответствующие им целевые функции, которые учитываются центром при выборе координирующих воздействий. Об этих задачах можно говорить как о задачах, модели которых формируются путем интеграции нескольких исходных моделей (объединением их общей координирующей схемой).

Различают следующие подходы к организации информационного обмена в процессе координации:

а) осуществляется реальный обмен информацией между центром и подсистемами на каждом шаге итеративного процесса;

б) после получения и анализа информации от подсистем центр осуществляет весь итеративный процесс самостоятельно и вырабатывает координирующий сигнал, который сообщается подсистемам.

П е р в ы й подход позволяет упростить координирующую задачу по сравнению со вторым и может быть реально связан с ограниченными вычислительными мощностями и пропускными способностями каналов связи (проще - передать координирующий сигнал и частные решения подсистем, чем передавать и перерабатывать всю информацию о совместной операционной области в подсистемах).

В т о р о й подход связан с более глубокими соображениями – с учетом факторов неопределенности. Действительно, в этом случае центр лишен возможности принимать решение по подсистемам, поскольку он не располагает полными сведениями об обстановке в них (эта обстановка на момент решения координирующей задачи может быть вообще неизвестна). Центр при этом самостоятельно ведет итеративный процесс и вырабатывает координирующий сигнал, учитывая неопределенность обстановки на основе прогнозирования ее развития с использованием методов принятия решений в неопределенной обстановке (вероятностных, игровых, в условиях неизвестности). Подсистемы не участвуют в итеративном процессе (или ограниченно привлекаются к нему), но принимают окончательные решения на основе знания координирующего сигнала и реально складывающейся обстановки.

Исторически первыми начали развиваться процедуры декомпозиции и иерархического выбора для вычислительных процедур оптимизации. Наиболее известен здесь декомпозиционный алгоритм Данцига-Вулфа [26, 59], разработанный для задач линейного программирования с матрицей ограничений блочной структуры. Дальнейшее развитие этих методов осуществлялось в направлении развития декомпозиционных процедур для нелинейных оптимизационных задач (алгоритм Розена [59]) и для задач с целочисленными и частично целочисленными переменными (алгоритм Бендерса [85]). Развиваются вычислительные процедуры, построенные на основе совместного применения идей декомпозиции и агрегирования.

Методы декомпозиции и координации, имеющие содержательный характер, т.е. связанные с проблемами управления в реальных иерархических системах развивались в работе [62], где была сделана попытка создания общей теории оптимизации иерархических систем. Большие возможности изучения содержательных задач координации открываются в связи с исследованиями Ю.Б.Гермейера [22] и Н.Н.Моисеева [63, 80] и их учеников, проводимыми с позиций теории игр с непротивоположными интересами. Координационный (согласованный) выбор, учитывающий активность и собственные интересы подсистем изучался в работах по теории активных систем В.Н.Бурковым [11]. Задачи координационного выбора в иерархических системах решались И. А. Багриновским [6] на основе введения межуровневого согласования по Нэшу.

Первыми, получившими широкую известность, оптимизационными алгоритмами с иерархической организацией вычислительного процесса и координацией решений подсистем на основе введения соответствующих координирующих сигналов были декомпозиционные алгоритмы Данцига-Вулфа [26] и Корнаи-Липтака [85]. В первом алгоритме использовалась интерпретация вычислительного процесса как процесса координации, построенного на основе модификации целевой функции с использованием механизма цен; во-втором - как процесса координации с модификацией пространства альтернатив, построенной на основе распределения ресурсов.

15.2. Координация с модификацией целевой функции

Алгоритм Данцига-Вулфа [26] строится на основе развития идей симплекс-метода, точнее метода обратной матрицы (модифицированный алгоритм симплекс-метода), для задач линейного программирования с блочно-диагональной структурой матрицы условий. В этом случае модель принятия решения (Δ, f), где

$$\Delta = \{\, x \in R^n \mid Ax = b,\ x \geq 0.\}; \quad f = c^T x .$$

Иначе, если задачу линейного программирования представить в виде:

$$x^* = \arg\max_{x \in \Delta} c^T x , \qquad (15.1)$$

где

$$\Delta = \{\, x \in R^n \mid Ax = b,\ x \geq 0. \}\,, \qquad (15.2)$$

то блочно-диагональная матрица условий A представима в виде:

$$A = \begin{Vmatrix} A_{o1} & A_{o2} & \ldots\ldots & A_{op} \\ A_1 & 0 & \ldots\ldots & 0 \\ 0 & A_2 & \ldots\ldots & \ldots \\ \ldots & \ldots & \ldots\ldots & \ldots \\ 0 & 0 & \ldots\ldots & A_p \end{Vmatrix} \qquad (15.3)$$

Матрицы A_i, i=1,...,p часто интерпретируются, как матрицы технологических условий выполнения работ i-ой подсистемы. Тогда подматрицу $\| A_{o1}\ A_{o2} \ldots\ A_{op} \|$ матрицы A интерпретируют, как матрицу условий использования общих ресурсов, необходимых для выполнения работ подсистем.

С учетом блочной структуры (15.3) матрицы условий исходную задачу можно представить в виде:

$$\sum_{i=1}^{p} c^{T}_{i}\, x_i \to \max, \qquad (15.4)$$

$$\sum_{i=1}^{p} A_{oi} x_i = b_o, \qquad (15.5)$$

$$A_i\, x_i = b_i,\ \ i = 1,\ldots, p, \qquad (15.6)$$

$$x_i \geq 0. \quad\ i = 1,\ldots, p. \qquad (15.7)$$

Здесь

c^{T}_{i}- n_i - мерный транспонированный вектор коэффициентов целевой функции i-ой подсистемы;

x_i - n_i - мерный вектор-решение i-ой подсистемы;

A_{oi}- $m_o \times n$ - мерная матрица глобальных условий (ресурсных ограничений) i-ой подсистемы;

b_o - m_o - мерный вектор глобальных ограничений;

A_i - $m_i \times n_i$ - мерная матрица условий (технологических ограничений) i-ой подсистемы;

b_i - m_i - мерный вектор ограничений.

Причем

$$\sum_{i=1}^{p} n_i = n, \quad \sum_{i=0}^{p} m_i = m,$$

где m и n - размерность матрицы А. Отметим, что к представленному виду можно свести любую задачу линейного программирования, тогда p = 2.

Ограничения (15.6), (15.7) задают p выпуклых многогранников Δ_i. Пусть x^*_{ij}, j=1,..., s_i - решения, соответствующие вершинам (крайним точкам) i-го выпуклого многогранника; s_i - количество вершин Δ_i. Тогда любое допустимое решение i-ой подсистемы x_i можно представить в виде:

$$x_i = \sum_{j=1}^{s_i} \lambda_{ij} x^*_{ij} , \qquad (15.8)$$

где коэффициенты λ_{ij} удовлетворяют условию

$$\sum_{j=1}^{s_i} \lambda_{ij} = 1,\ \lambda_{ij} \geq 0,\ i=1,\ldots, p, \qquad (15.9)$$

Введем следующее предположение.

__Предположение 1__. Пусть известно множество X^* всех вершин, выпуклых многогранников Δ_i , i=1,...,p, задаваемых условиями (15.6), (15.7).

$X^* = \{ x^*_{ij} \}$, i = 1,..., p, j = 1,..., s_i.

Тогда, подставляя выражения (15.8), вместо x_i в исходную оптимизационную задачу (15.4)-(15.7), получим

$$\sum_{i=1}^{p} \sum_{j=1}^{s_i} c^T_i (\lambda_{ij} x^*_{ij}) \to \max, \qquad (15.10)$$

$$\sum_{i=1}^{p} \sum_{j=1}^{s_i} A_{oi} (\lambda_{ij} x^*_{ij}) = b_o , \qquad (15.11)$$

$$\sum_{i=1}^{p} \sum_{j=1}^{s_i} A_i (\lambda_{ij} x^*_{ij}) = b_i,\ i = 1,\ldots, p, \qquad (15.12)$$

$$\lambda_{ij} \geq 0,\ i=1,\ldots, p,\ j=1,\ldots, s_i, \qquad (15.13)$$

Обозначим

$$\varphi_{ij} = c^T_i x^*_{ij},\ \alpha_{ij} = A_{oi} x^*_{ij}, \qquad (15.14)$$

где φ_{ij} - значение целевой функции i-ой подсистемы, вычисленное на решении x^*_{ij}; α_{ij} - вектор глобальных ресурсов, необходимых для реализации решения x^*_{ij}. Кроме того, в (15.12) будем иметь в виду, что $A_i x^*_{ij} = b_i$, i = 1,..., p.

Тогда задачу (15.10)-(15.13) можно переписать в виде

$$\sum_{i=1}^{p} \sum_{j=1}^{s_i} \varphi_{ij} \lambda_{ij} \to \max, \qquad (15.15)$$

$$\sum_{i=1}^{p} \sum_{j=1}^{s_i} \alpha_{ij} \lambda_{ij} = b_o , \qquad (15.16)$$

$$\sum_{j=1}^{s_i} \lambda_{ij} = 1, \;\; i = 1,..., p, \qquad (15.17)$$

$$\lambda_{ij} \ge 0, \;\; i=1,..., p, j=1,\dots, s_i, \qquad (15.18)$$

Таким образом, исходная задача (15.4)-(15.7) в условиях предположения 1 сведена к задаче (15.15)-(15.18) с переменными λ_{ij}, число которых хотя и конечно, но может быть достаточно велико. В то же время, если исходная задача имела ($m_o + m_1 + m_2 + \cdot\cdot\cdot + m_p$) ограничений, то задача (15.15)-(15.18) имеет всего (m_o+p) ограничений. Матрица условий задачи (15.15)-(15.18) имеет вид:

$$\left\| \begin{array}{ccc|ccc|c|ccc} \alpha_{11}, & \alpha_{12}, \dots, & \alpha_{1s} & \alpha_{21}, & \alpha_{22}, \dots, & \alpha_{2s} & \cdots\cdots & \alpha_{p1}, & \alpha_{p2}, \dots, & \alpha_{ps} \\ \hline 1 & 1 \dots\dots & 1 & 0 & 0 \dots\dots & 0 & \cdots\cdots & 0 & 0 \dots\dots & 0 \\ 0 & 0 \dots\dots & 0 & 1 & 1 \dots\dots & 1 & \cdots\cdots & 0 & 0 \dots\dots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots\cdots & \cdots & \cdots & \cdots \\ 0 & 0 \dots\dots & 0 & 0 & 0 \dots\dots & 0 & \cdots\cdots & 1 & 1 \dots\dots & 1 \end{array} \right\|$$

Рис. 15.2.

Согласно методу обратной матрицы (см. 3.5) решение задачи (15.15)- (15.18) оптимально, если выполняется следующее условие:

$$\delta_k = \min_{i,j} (\pi^{т} a^{o}_{ij} - \varphi_{ij}) \ge 0, \qquad (15.19)$$

где

$\pi^{т}$ - транспонированный вектор двойственных оценок условий (15.16), (15.17):

$$\pi^{т} = \| \; \pi^{т}_{k} \,\vert\, \pi_1 \,\vert\, \dots\dots \,\vert\, \pi_p \; \| \qquad (15.20)$$

здесь $\pi^{т}_{k}$ - транспонированный вектор двойственных оценок условий (15.16); π_i - двойственная оценка i-го условия вида (15.17);

a^o_{ij}- обобщенный вектор условий задачи (15.15) - (15.18) с матрицей условий, изображенной на рис.15.2; a^o_{ij} имеет вид:

$$a^o_{ij} = \left\| \begin{array}{c} a_{ij} \\ ----- \\ e_i \end{array} \right\| \qquad (15.21)$$

где e_i - вектор, i-я компонента которого равна 1, а остальные компоненты равны 0.

Преобразуем условие оптимальности (15.19) с учетом (15.20), (15.21):

$$\delta_k = \min_{ij} (\pi^T a^o_{ij} - \varphi_{ij}) = \min_{ij} (\pi^T_k a_{ij} + \pi_i - \varphi_{ij}) =$$

$$= \min_i (\pi_i - \max_j (\varphi_{ij} - \pi^T_k a_{ij})) \geq 0,$$

Здесь задача max $(\varphi_{ij} - \pi^T_k a_{ij})$, j=1,..., s_i представляет из себя самостоятельную оптимизационную задачу i-ой подсистемы. Действительно, с учетом (15.14) ее можно представить как

$$\max_j (\varphi_{ij} - \pi^T_k a_{ij})) = \max_j (c^T_i x_{ij} - \pi^T_k A_{oi} x_{ij}) = \qquad (15.22)$$

$$\max_j ((c^T_i - \pi^T_k A_{oi}) x_{ij}) = \max_{x_i \in \Delta_i} ((c^T_i - \pi^T_k A_{oi}) x_i) ,$$

где Δ_i определяется ограничениями (15.6), (15.7) и задает многогранник допустимых решений i-ой подсистемы.

Сравнивая полученную задачу оптимизации решений в подсистемах с исходной, можно отметить, что целевая функция подверглась модификации. Действительно, если исходная целевая функция i-ой подсистемы имела вид $(c^T_i x_i)$, то целевая функция подсистем в (15.22) имеет вид $(c^T_i - \pi^T_k A_{oi}) x_i$, т.е. из коэффициентов целевой функции вычитается стоимость общих ресурсов, необходимых для реализации решения x_i; π^T_k интерпретируется как стоимость ресурсов в единицах глобального критерия ("механизм цен").

На каждом шаге решения координирующей задачи решается p оптимизационных задач подсистем, решения которых служат информационной основой для координирующей задачи следующего шага. В этих условиях "Предположение 1" больше не является необходимым, т.к. выбор наилучшей на данном шаге крайней точки многогранника в соответствии с (15.22) эквивалентен решению оптимизационной задачи

$$\max_{x_i \in \Delta_i} ((c^T_i - \pi^T_k A_{oi})\, x_i) \,.$$

Такой метод соответствует процедуре "генерации столбца" [59]. Сама же координирующая задача имеет вид (15.15)-(15.18) с той разницей, что s_i соответствует количеству решений, которые i-ая подсистема сообщила координатору на некотором шаге решения координирующей задачи.

Отметим, что оптимальное решение x^*_i задачи в целом, как правило, не является простым объединением решений подсистем. Из (15.8) следует, что окончательное решение подсистемы является линейной комбинацией ряда промежуточных решений x^*_{ij} с коэффициентами λ^*_{ij}, найденными в результате решения координирующей задачи,

$$x^*_i = \sum_{j=1}^{s_i} \lambda^*_{ij}\, x^*_{ij} \,.$$

Тогда оптимальное решение не обязательно соответствует некоторой вершине многогранника, описывающего допустимые решения i-ой подсистемы, но, как правило, соответствует некоторой внутренней точке Δ_i. В этом смысле рассматриваемый алгоритм декомпозиции не является средством полной децентрализации процесса принятия решений. Более удачно определение Дж. Данцига [26], как "централизованное планирование без использования в центре полной информации".

Рассмотрим алгоритм координационного принятия решения в иерархической системе.

На некоторой r-ой итерации подсистемы решают оптимизационные задачи (Δ_i, f_i), где Δ_i описывается ограничениями (15.6), (15.7); $f_i = (c^T_i - \pi^T_{ki})\, x_i$. Центр на основе найденных в подсистемах решений x^*_{ir} и значений целевой функции $\varphi_{ir} = f_i(x^*_{ir}) = c^T_i x_{ir}$ строит координирующую задачу, используя при этом матрицы $A_{oi,}$ i=1,.. ..,p. В результате решения координирующей задачи вычисляются координирующие сигналы $\pi^T_{ki} = \pi^T_k A_{oi}$, которые сообщаются в подсистемы. Подсистемы на основе полученного из Центра координирующего сигнала π^T_{ki} производят модификацию целевой функции по правилу $(c^T_i - \pi^T_{ki})x_i$, и находят новые оптимальные решения x^*_{ir}, которые сообщаются в Центр. И т.д. Процесс заканчивается, когда координирующий сигнал, найденный в результате решения задачи Центра перестает изменяться.

Обмен информацией между подсистемами и Центром можно характеризовать следующей схемой

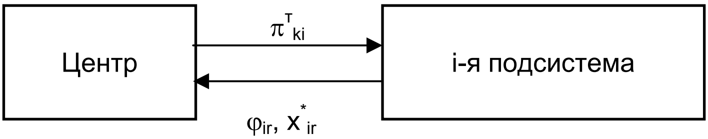


Рис. 15.3.

<u>Алгоритм Данцига Вулфа.</u>

Алгоритм носит итерационный характер. Каждая итерация состоит из нескольких шагов. Рассмотрим содержание действий, производимых на каждом шаге.

0). Задается начальный номер итерации r = 1. Центр сообщает всем подсистемам одинаковый координирующий сигнал

$\pi^{T}_{ki} = 0^{T}$, i = 1,...,p.

1). Подсистемы решают оптимизационные задачи вида

$$(c^{T}_{i} - \pi^{T}_{ki})\, x_i \to \max$$
$$A_i\, x_i = b_i,$$
$$x_i \ge 0.$$

В результате решения данных задач находятся оптимальные решения подсистем на r-ой итерации x^{*}_{ir}. В соответствии с полученными решениями производится вычисление значений φ_{ir}, $\varphi_{ir} = c^{T}_{i} x^{*}_{ir}$. Значения φ_{ir}, x^{*}_{ir} сообщаются в Центр.

2). Строится координирующая задача в виде

$$\sum_{i=1}^{p} \sum_{j=1}^{r} \varphi_{ij}\, \lambda_{ij} \to \max, \tag{15.23}$$

$$\sum_{i=1}^{p} \sum_{j=1}^{r} \alpha_{ij}\, \lambda_{ij} = b_o\ , \tag{15.24}$$

$$\sum_{j=1}^{r} \lambda_{ij} = 1, \quad i = 1,..., p, \tag{15.25}$$

$$\lambda_{ij.} \ge 0, \quad i=1,..., p,\ j=1,\dots, r, \tag{15.26}$$

Здесь вектор a_{ij} вычисляется как $\alpha_{ij} = A_{oi} x^*_{ij}$, i=1,...,p, j=1,...,r, где $\{x^*_{ij}\}$, i=1,...,p, j=1,...,r - множество решений, переданных подсистемами в Центр на r итерациях. Данная задача является задачей линейного программирования и может решаться с использованием симплекс-метода (метод обратной матрицы (см.3.5)).

Если на первом шаге решения этой задачи устанавливается оптимальность текущего решения (т.е. ни одна из переменных λ_{ir}, i=1,...,p, соответствующих решениям подсистем x^*_{ir}, найденным на r-ой итерации), не вводится в базис, то, следовательно, координирующий сигнал не будет пересчитываться (останется без изменений). Это означает, что найдено оптимальное решение исходной задачи - производится переход на шаг 5.

3). Производится расчет координирующих сигналов. В нулевой строке симплекс-таблицы, соответствующей оптимальному решению координирующей задачи (15.23)-(15.26) (см. алгоритм в 3.5), содержатся компоненты транспонированного вектора двойственных оценок π^T. Первые m_o компонент вектора π^T (m_o - количество глобальных ограничений в (15.5) образуют вектор π^T_k. Тогда индивидуальный координирующий сигнал i-ой подсистемы рассчитывается как

$$\pi^T_{ki} = \pi^T_k A_{oi}.$$

Значения координирующих сигналов π^T_{ki}, i=1,...,p сообщаются в подсистемы.

4). Изменяется номер итерации r = r + 1. Производится переход на шаг 1.

5). Формируются оптимальные решения подсистем по правилу

$$x^*_i = \sum_{j=1}^{r} \lambda^*_{ij} x^*_{ij} .$$

Здесь x^*_{ij}, j = 1,...,r решения i-ой подсистемы, которые сообщались Центру на r итерациях решения задачи; λ^*_{ij}, i = 1,...,p, j = 1,...., r - компоненты оптимального решения координирующей задачи (15.23)-(15.26).

Замечание 1.

Размерность координирующей задачи (15.23)-(15.26) постоянно увеличивается, т.к. на каждой итерации в нее добавляется p новых столбцов, соответствующих решениям подсистем, найденным на данной итерации. Следует отметить, что подсистемы на некоторой r-ой итерации (r > 1) могут сообщать в Центр решения, которые там уже имеются. В этом случае такие решения можно не включать в координирующую задачу.

<u>Замечание 2.</u>

На первой итерации подсистемы сообщают в Центр решения, соответствующие наилучшему значению целевой функции подсистемы на Δ_i (т.к. $\pi^T_{ki} = 0^T$, i=1,...,p). В этих условиях (при r = 1) решение координирующей задачи (15.23)-(15.26) довольно часто не существует, т.к., как правило, не хватает общих ресурсов на реализацию таких оптимальных решений подсистем. Поэтому для решения координирующей задачи целесообразно использовать М-метод решения задач линейного программирования (см. 3.6), при использовании которого исходная задача (15.23)-(15.26) погружается в более широкую М-задачу, исходное опорное решение которой очевидно - это дополнительные переменные, количество которых равно (m_o + p). Использование дополнительных переменных при этом штрафуется (М - размер штрафа), вследствие чего дополнительные переменные в процессе решения координирующей задачи выводятся из базиса.

15.3. Координационное планирование на основе перераспределения ресурсов

Аналогично тому, как это рассматривалось в предыдущем параграфе будем интерпретировать задачу принятия решения в иерархической системе, как задачу независимого планирования действий подсистем, которые требуют при реализации своих индивидуальных решений привлечения некоторых общих для все подсистем ресурсов. В отличие от декомпозиции Данцига-Вулфа, где координация действий подсистем достигалась за счет модификации целевой функции, а координирующие сигналы имели смысл стоимости общих ресурсов, в декомпозиции Корнаи-Липтака [85] в качестве координирующих сигналов рассматривается количество глобальных ресурсов, выделяемых подсистеме, в связи с чем модификации подвергаются ограничения, описывающие потребности подсистемы в общих ресурсах, и, как следствие этого, модифицируется множество допустимых решений подсистемы.

Перепишем задачу (15.4)-(15.7) в виде:

$$\sum_{i=1}^{p} c^T{}_i\, x_i \to \max, \qquad (15.27)$$

$$\sum_{i=1}^{p} A_{oi} x_i \le b_o, \qquad (15.28)$$

$$A_i\, x_i \le b_i, \;\; i = 1,..., p, \qquad (15.29)$$

$$x_i \ge 0. \quad\;\; i = 1,..., p. \qquad (15.30)$$

Обозначим $\pi_i = \pi_i(x_i)$,как

$$\pi_i = A_{oi}\, x_i. \qquad (15.31)$$

Тогда задачу (15.27)-(15.30) можно представить в виде совокупности оптимизационных задач.

Задача Центра:

$$f(\pi) = \sum_{i=1}^{p} f_i(\pi_i) = \sum_{i=1}^{p} c^T{}_i\, x_i(\pi_i) \to \max, \qquad (15.32)$$

$$\Delta_o: \quad \sum_{i=1}^{p} \pi_i \le b_o\,. \qquad (15.33)$$

Задача i - ой подсистемы

$$c^T{}_i\, x_i \to \max, \qquad (15.34)$$

$$A_{oi} x_i \le p_i, \qquad (15.35)$$

$$\Delta_i: \quad A_i\, x_i \le b_i, \qquad (15.36)$$

$$x_i \ge 0. \qquad (15.37)$$

Здесь ограничения (15.36), (15.37) описывают технологические условия функционирования i - ой подсистемы; (15.35) представляют собой ресурсные ограничения. При этом π_i - доля глобального ресурса, который выделяется Центром i-ой подсистеме; π_i - выступает в качестве координирующего сигнала. Так как π_i в задаче подсистемы (15.34) - (15.37) является подвектором вектора ограничений i-ой подсистемы, то, следовательно, модификации подвергается множество допустимых альтернатив $\Delta_i = \Delta_i(\pi_i)$.

Задача Центра (15.32), (15.33) заключается в нахождении оптимального распределения общего ресурса. Решение такой задачи усложняется тем, что функции $x_i(\pi_i)$ (а, следовательно, и $f(\pi)$) заданы алгоритмически. Действительно, для нахождения значений $x_i(\pi_i)$ необходимо решить задачи подсистем (15.34) - (15.37). Применение той или иной вычислительной схемы при максимизации (15.32) порождает соответствующий алгоритм, построенный на основе декомпозиции. При использовании алгоритма Корнаи-Липтака данная задача сводится к максиминной задаче, которая исследуется методами теории игр.

Анализ задачи (15.32), (15.33) показывает целесообразность передачи ресурсов той подсистеме, которой соответствуют максимальные значения целевой функции $f_i(\pi_i) = c^T_i\, x_i(\pi_i)$. Вместе с тем, учитывая связь целевых функций прямой и двойственной задачи, можно записать

$$f_i(\pi_i) = c^T_i\, x_i(\pi_i) = y^T_i(\pi_i)\, \mu_i, \qquad (15.38)$$

где

$y^T_i(\pi_i)$ - транспонированный вектор всех ограничений задачи i-ой подсистемы;

$$y^T_i(\pi_i) = \| \, \pi_i \,\vdots\, b_i \, \|^T; \qquad (15.39)$$

μ_i - решение задачи, двойственной по отношению к задаче (15.34) - (15.37); соответственно (15.39) решение двойственной задачи - вектор μ_i можно представить в виде

$$\mu_i = \| \, \mu_{\pi i} \,\vdots\, \mu_{bi} \|^T. \qquad (15.40)$$

Здесь значения двойственных переменных $\mu_{\pi i}$ соответствуют общим ресурсам π_i, которые передаются Центром i-ой подсистеме, и характеризуют эффективность их использования в единицах критерия. Тогда (15.38) можно переписать

$$f_i(\pi_i) = \pi^T_i\, \mu_{\pi i} + b^T_i\, \mu_{bi}, \qquad (15.41)$$

Целевая функция глобальной задачи (Центра) (15.32) с учетом выражений (15.41) имеет вид

$$\sum_{i=1}^{p} y^T_i(\pi_i)\, \mu_i = \sum_{i=1}^{p} \pi^T_i\, \mu_{\pi i} + \sum_{i=1}^{p} b^T_i\, \mu_{bi} = \sum_{i=1}^{p} \mu^T_{\pi i}\, \pi_i + C.$$

Тогда задача Центра:

$$(\mu, \pi) = \sum_{i=1}^{p} \mu^T_i\, y_i(\pi_i) \geq \sum_{i=1}^{p} \mu^T_{\pi i}\, \pi_i \to \max, \qquad (15.42)$$

$$\Delta_o: \qquad \sum_{i=1}^{p} \pi_i \leq b_o\,. \qquad (15.43)$$

Обобщенно данную задачу можно записать как

$$(\mu, \pi^*) = \max_{\pi \in \Delta_o} (\mu, \pi) \qquad (15.44)$$

Значения $\mu_{\pi i}$ в (15.42) находятся в результате решения в подсистемах задач, двойственных по отношению к задаче (15.34) - (15.37). Такие задачи имеют вид

$$\| \pi^{T}_{i} \mid b^{T}_{i} \| \, \mu_i \to \min, \tag{15.45}$$

$$\Delta^{c}_{i}: \quad \| A^{T}_{oi} \mid A^{T}_{i} \| \, \mu_i \geq c_i, \tag{15.46}$$

$$\mu_i \geq 0. \tag{15.47}$$

Обобщенно последнюю задачу (по аналогии с (15.44)) можно записать как

$$(\mu^{*}_{i}, \pi_i) = \min_{\mu_i \in \Delta^{c}_{i}} (\mu_i, \pi_i), \; i = 1,...,p . \tag{15.48}$$

Объединяя задачу (15.44) и задачи (15.48), можно заключить, что оптимальное в целом решение ищется как

$$(\mu^{*}, \pi^{*}) = \max_{\pi \in \Delta_o} \min_{\mu \in \Delta^{c}} (\mu, \pi) . \tag{15.49}$$

Здесь

$$\Delta^{c} = \prod_{i=1}^{p} \Delta^{c}_{i} .$$

Задачи подсистем (15.45)-(15.47) относятся к классу задач линейного программирования и могут быть решены на основе симплекс-метода. Здесь следует отметить, что использование метода обратной матрицы для решения прямой задачи (15.34) - (15.37) позволяет получать решение и двойственной задачи (15.45) - (15.47).

Анализ задачи Центра показывает, что целесообразно отдавать ресурсов π_i больше той подсистеме, где эффективность их использования $\mu_{\pi i}$ выше. Тогда очевидное решение такой задачи имеет вид

$$\pi_{ij} = \begin{cases} 0, & \mu_{\pi ij} < \max\limits_{i} \mu_{\pi ij} \\ b_{oj} , & \mu_{\pi ij} = \max\limits_{i} \mu_{\pi ij} \end{cases}$$

Непосредственное использование последнего выражения для распределения глобальных ресурсов делает, как правило, вычислительный процесс неустойчивым, поэтому используются другие схемы решения задачи. Алгоритм Корнаи-Липтака является исторически первым способом решения такой задачи, в основе которого лежит рассмотрение итеративной процедуры, как последовательности партий игры двух лиц. В качестве первого игрока выступает Центр, а в качестве второго выступают подсистемы, которые независимо друг от друга решают свои оптимизационные задачи. Сходимость процесса обеспечивается использованием итеративной процедуры Брауна-Роббинсон (см. 13.3.3). При этом Центр решает задачу

$$(\mu, \pi^*(\mu)) = \max_{\pi\in\Delta_o} (\mu, \pi)$$

в подсистемах решается задача

$$(\mu^*(\pi), \pi) = \min_{\mu\in\Delta^c} (\mu, \pi).$$

Алгоритм носит итерационный характер, каждая итерация которого состоит из нескольких шагов.

Начальная итерация (1-я).

1). Выбирается произвольное распределение глобальных ресурсов, например, как $\pi^*_{ij} = b_{oj} / p$, $j=1,...,m_o$.

2). В результате решения задач (15.45)-(15.47) вычисляется μ^*.

3). Полагаем $\mu[1] = \mu^*$; $k = 2$. Задается критерий завершения итерационного процесса $\varepsilon > 0$.

Произвольная итерация (k-ая):

1). В результате решения задачи (15.42), (15.43) находится значение $\pi^* = \pi^*(\mu[k-1])$.

2). Вычисляется значение

$\pi[k] = (k-1)/k\ \pi[k-1] + 1/k\ \pi^*$.

3). В результате решения задач вида (15.45) - (15.47) находится значение $\mu^* = \mu^*(\pi[k]))$.

4). Вычисляется значение

$\mu[k] = (k-1)/k\ \mu\ [k-1] + 1/k\ \mu^*$.

5). Если $\| \mu[k] - \mu[k-1] \| \leq e$, то процесс окончен.

Если $\| \mu[k] - \mu[k-1] \| > e$, то $k = k + 1$; производится переход на шаг 1.

Последовательность значений $(\mu^*[j], \pi^*[j])$ сходится к седловой точке исходной задачи (15.49). Вместе с тем сходимость данной процедуры достаточно медленная. В ряде работ рассматриваются другие схемы решения данной координационной задачи, обладающие более высокой сходимостью.

15.4. Релаксация в задачах координации

Релаксация, заключающаяся во временном отбрасывании части ограничений, особенно эффективна в ситуациях со специфической структурой условий или, когда заранее известно, что часть ограничений несущественна при оптимальном решении. В ряде случаев часть ограничений (R) задачи удовлетворяет некоторым теоретическим условиям, позволяющим использовать эффективные методы оптимизации. Вместе с тем в задаче присутствуют и другие ограничения (P), не соответствующие таким условиям. Тогда разумная стратегия решения такой задачи состоит во временном отбрасывании P ограничений и решении релаксированной R - задачи. Если такая R - задача неразрешима, то неразрешима и исходная задача. Если она разрешима, и полученное решение удовлетворяет P - ограничением, то решение оптимальное для R - задачи будет и в целом оптимально. В противном случае необходимо ввести в R-задачу одно или более из P ограничений и повторить процедуру.

Итак, пусть имеется задача

$$x^* = \mathrm{Arg}\max_{x\in\Delta} f(x)\ ,\ \Delta = \{x\in R^n \mid g_i(x) \le b_i\ ,\ i \in M\}$$

Релаксированная R - задача заключается в учете ограничений только из множества R, где $R\subseteq M$. Предположим, что существует конечный максимум f(x), который достигается на решении x^R, тогда релаксационная стратегия представляет собой итеративный процесс, заключающийся в выполнении последовательности следующих шагов.

0) Назначим $F = +\infty$; выбирается подмножество ограничений $R\subseteq M$ такое, что f(x) ограничена на $\Delta^R = \{x\in R^n \mid g_i(x) \le b_i\ ,\ i \in R\}$.

1) Решается R-задача. Если она неразрешима, то и исходная задача неразрешима, в противном случае x^R - оптимальное решение такой задачи.

2) Решение x^R проверяется на допустимость по отношению к оставшимся ограничениям. Если $g_i(x^R) \le b_i$, $i\in M\backslash R$, то x^R - оптимальное решение исходной задачи в целом.

3) Пусть $S\subseteq M\backslash R$, такое, что $\exists k\in S$, для которого не выполняется k-е ограничение при x^R (т.е. $g_k(x^R) > b_i$). Введем подмножество $V\subseteq R$, для которого ограничения выполняются как строгие неравенства $V = \{\ j\in R \mid g_j(x^R) < b_i\ \}$.

4) Тогда,
- если выполняется $f(x^R) = F$, то $R = R \cup S$;
- если выполняется $f(x^R) < F$, то $R = R \cup S \backslash V$; $F=f(x^R)$.
Осуществляется переход к шагу 1.

Таким образом, в процедуре добавляется одно или более невыполненных ограничений, если F не изменяется, и добавляются невыполненные ограничения и изымаются несущественные ограничения, если F уменьшается. Данная процедура за конечное число шагов либо устанавливает неразрешимость исходной задачи, либо приводит к оптимальному решению.

Описанная процедура достаточно эффективна в условиях специфической структуры ограничений и служит для формального обоснования эвристики, используемой при решении задач координации. Действительно, если ограничения имеют блочно-диагональной вид, то, рассматриваемая релаксированная задача будет иметь блочную структуру (определяемую задачами подсистем). В этом случае решение рассматриваемой релаксированной задачи сводится к решению p независимых задач, для которых впоследствии учитываются связующие ограничения. В других случаях ряд ограничений может удовлетворять определенным условиям, которые позволяют использовать для этой части ограничений эффективные методы оптимизации решения, и использование идей релаксации для таких задач также оказывается весьма полезным.

15.5. Координационное планирование операций управления КА

Функционирование автоматизированной системы управления КА характеризуется, с одной стороны, большим разнообразием задач, решаемых объектами, а с другой - взаимной зависимостью их программ управления, которая обусловлена единым для всех объектов комплексом технических средств, привлекаемых к управлению. Указанные особенности функционирования системы необходимо учитывать при решении задач автоматизации управления и, в частности, планирования на основе разработки комплекса частных моделей планирования для отдельных КА (примеры таких моделей рассматривались в 12.4) и модели координации планов, которую рассмотрим здесь.

Первым шагом построения модели координации является формализация конфликтов между подсистемами в удобном для анализа виде. Представим общий план в виде $u = \|u_1,...,u_\nu,...,u_n\|$, где u_ν - план управления ν-м КА. Плану u_ν однозначно соответствует упорядоченное множество запланированных действий $\|y_i, i\in I_\nu\|$. В плане u_ν содержатся компоненты, соответствующие операциям управления, запланированным предварительно в подсистемах (на основе моделей планирования операций отдельных КА см.1.4). Тогда $u\in\Delta$, где Δ - множество допустимых планов всей системы в целом, если $u_\nu\in\Delta_\nu$, $\nu\in N = \{1,...,n\}$, и планы отдельных КА u_ν не конфликтуют между собой по привлекаемым в процессе реализации этих планов техническим средствам управления.

Можно различать следующие ситуации, приводящие к конфликтам операций управления:

а) конфликты локализованы на сравнительно небольшом интервале времени и определяются ограниченной пропускной способностью средств управления. Это, например, ситуации, когда:

- операции проводятся в одно и то же время и требуют привлечения одного и того же технического средства управления;
- операции проводятся в разное время, но требуют привлечения технического средства, которое не успевает перестроиться после выполнения одной операции для выполнения другой;
- совместная работа средств невозможна по условиям технической совместимости проводимых работ;

и т.д.

б) ситуации, когда имеются ограничения ресурсного типа, которые определяются техническими возможностями и особенностями функционирования средств управления и проявляются на длительных временных промежутках, сравнимых с интервалом планирования. Например,

- ограниченная длительность непрерывной работы средства управления;
- ограничения по информационной загрузке каналов связи при передаче данных между пунктами управления;
- ограниченный суммарный расход некоторых видов ресурсов на интервале управления

и др.

Ситуации первого типа имеют локальный характер, и описываемые ими конфликты должны анализироваться в каждый момент времени. Рассмотрим особенности решения задачи, связанной с развязыванием конфликтов такого рода. Основное содержание данной релаксированной задачи определяется конфликтными ситуациями, имеющими место между планами управления различными КА, которые определяются конфликтами действий, составляющих эти планы. Указанные конфликты можно описать рефлексивным бинарным отношением R_k или соответствующим ему неориентированным графом G_k без петель и кратных ребер, где вершинам соответствуют действия, а ребрам соответствуют существующие между действиями конфликты. Пусть А - матрица инциденций такого графа, тогда план всей системы в целом, составленный из планов выполнения операций управления КА, $u = \|u_1,...,u_\nu,...,u_n\|$ будет допустимым планом, если

$$\sum_{\nu=1}^{n} A_\nu\, u_\nu \le e,$$

$$u_\nu \in \Delta_\nu, \quad \nu \in N,$$

где A_ν подматрица матрицы инциденций $A = \| A_1, ..., A_n \|$, соответствующая плану u_ν управления ν-м КА; e - вектор, все компоненты которого равны единице; Δ_ν - множество допустимых планов ν-го КА.

Проводя анализ графа G_k, можно заключить, что множество его вершин разбивается на подмножества, соответствующие полным подграфам графа G_k (на клики). Из каждой такой клики можно выбрать только одну вершину, так как все остальные вершины данного подмножества (клики) с ней конфликтуют.

Данную конфликтную обстановку можно описать с помощью гиперграфа H_k - обобщенного неориентированного графа, вершины которого соответствуют вершинам G_k, а ребрами являются подмножества конфликтующих вершин. Сопоставив теперь каждому ребру гиперграфа H_k клику в графе конфликтов G_k, условия совместности планов u_ν, можно записать с использованием матрицы A^c инциденций гиперграфа H_k в виде $A^c u \le e$. В задачах планирования операций управления размерность A^c, как правило, существенно меньше размерности матрицы А.

Таким образом, условия развязывания конфликтных ситуаций можно формально описать совокупностью линейных алгебраических неравенств:

$$A^c\, u = \sum_{\nu=1}^{n} A^c{}_\nu\, u_\nu \le e,$$

В целом задачу оптимального планирования работы средств можно представить в виде

$$u^* = \arg\ \mathrm{opt}\ \{\ g(u)\ |\ \sum_{\nu=1}^{n} A^c_\nu\ u_\nu \le e,\ u_\nu \in \Delta_\nu,\ \nu \in N\ \}, \qquad (15.50)$$

где $u^* = \| u^*_1,\ldots,u^*_\nu,\ldots,u^*_n \|$ - оптимальное решение;

$g(u)$- глобальная целевая функция системы в целом;

A^c_ν - ν-я подматрица матрицы $A^c = \| A^c_1,\ldots, A^c_\nu,\ldots,A^c_n \|$;

opt - экстремальная характеристика $g(u)$, используемая при нахождении оптимального плана операций; в дальнейшем для определенности будем полагать, что opt = max;

Δ_ν- множество допустимых планов ν - го КА.

Полагая, что качество функционирования комплекса в целом достигается независимым (в смысле решения целевых задач) функционированием объектов, целевую функцию $g(u)$ можно представить в виде

$$g(u) = \sum_{\nu=1}^{n} \gamma_\nu g_\nu(u_\nu),$$

т.е. полагать сепарабельной относительно целевых функций КА.

Здесь γ_ν относительная важность ν-го объекта; $\forall\gamma_\nu \ge 0$; $\Sigma\gamma_\nu = 1$. Значения γ_ν вычисляются в результате анализа целей и задач, решаемых ν-м КА, и представляют, по-существу, результат формализации решающего правила в задаче векторной оптимизации, где векторный критерий образован целевыми функциями отдельных КА.

Тогда задачу (15.50) можно переписать в виде

$$u^* = \arg\max\ \{ \sum_{\nu=1}^{n} \gamma_\nu g_\nu(u_\nu)\ |\ \sum_{\nu=1}^{n} A^c_\nu\ u_\nu \le e,\ u_\nu \in \Delta_\nu,\ \nu \in N\ \}, \qquad (15.51)$$

Обозначим $|\Delta_\nu| = S_\nu$, $\nu \in N$ и сопоставим каждому j-му варианту допустимого плана $u_{\nu j}$ булеву переменную $\delta_{\nu j}$. Тогда задачу (15.51) можно представить как

$$\sum_{\nu=1}^{n} \sum_{j=1}^{S_\nu} \gamma_\nu\ g_{\nu j}\,\delta_{\nu j} \to \max, \qquad (15.52)$$

$$\sum_{\nu=1}^{n} \sum_{j=1}^{S_\nu} \alpha_{\nu j}\ \delta_{\nu j} \le e, \qquad (15.53)$$

$$\sum_{j=1}^{S_\nu} \delta_{\nu j} = 1,\ \nu \in N, \qquad (15.54)$$

$$\delta_{\nu j} \in \{0,1\},\quad \nu \in N,\ j = 1,\ldots,S_\nu. \qquad (15.55)$$

Здесь $g_{\nu j} = g_\nu(u_{\nu j})$; $\alpha_{\nu j} = A^c_\nu u_{\nu j}$.

Таким образом, нелинейная в общем случае задача (15.51), с учетом проведения соответствующих расчетов для $g_{\nu j}$ и $\alpha_{\nu j}$, сведена к задаче линейного булевого программирования. Особенность последней задачи заключается в том, что элементы матрицы $A^c_\nu = \{a_{ij}\}$, $a_{ij} \in \{0,1\}$, тогда и компоненты вектора $\alpha_{\nu j}$ также принимают значение 0 или 1. Последнее объясняется тем, что конфликты между действиями ν-го объекта полагаются развязанными, т.е. план $u_{\nu j}$ допустим с точки зрения использования средств управления. В этих условиях все допустимые решения, определяемые условиями (15.53)-(15.55), соответствует крайним точкам многогранника, задаваемого ограничениями (15.53), (15.54) при $\delta_{\nu j} \geq 0$. Множество таких точек связно. Тогда, используя алгоритм симплексного типа, анализирующий целочисленность вершины, в которую осуществляется переход, можно из исходной целочисленной вершины достичь оптимальной вершины, которая и будет решением задачи (15.52)-(15.55).

Поскольку все допустимые решения задачи (15.52)-(15.55) являются крайними точками многогранника (15.53), (15.54), $\delta_{\nu j} \geq 0$, $\forall(\nu,j)$, то для них справедливы соотношения двойственности математического программирования. Тогда критерием оптимальности некоторого решения $\delta = \left\| \delta_{\nu j} \right\|$ является условие

$$\min_{\nu,j} \left((\pi_k, \alpha_{\nu j}) - \gamma_\nu g_{\nu j} + \pi_\nu \right) \geq 0, \qquad (15.56)$$

где $(\pi_k, \alpha_{\nu j})$ - скалярное произведение векторов π_k и $\alpha_{\nu j}$;

π_k - вектор двойственных оценок условий (15.53);

π_ν - двойственная оценка ν - го условия вида (15.54).

Выражение (15.56) можно преобразовать:

$$\min_{\nu,j} \left((\pi_k, \alpha_{\nu j}) - \gamma_\nu g_{\nu j} + \pi_\nu \right) = \min_{\nu} \left(\pi_\nu - \max_{j}(\gamma_\nu g_{\nu j} - (\pi_k, \alpha_{\nu j})\right) =$$
$$\min_{\nu} \left(\pi_\nu - \gamma_\nu \max_{u_\nu \in \Delta_\nu} (g_\nu(u_\nu) - (\pi_k, A^c_\nu u_\nu)\right) \geq 0, \qquad (15.57)$$

Здесь выражение $\max (g_\nu(u_\nu) - (\pi_k, A^c_\nu u_\nu))$, $u_\nu \in \Delta_\nu$ представляет собой самостоятельную оптимизационную задачу планирования операций управления ν-го КА. Варианты таких моделей (статических и динамических) рассматривались ранее в 12.4. Из выражения (15.57) ясно, что скалярное произведение $(\pi_k, A^c_\nu u_\nu)$ является механизмом, посредством которого координатор воздействует на подсистемы (модели планирования операций управления отдельных объектов), а именно через модификацию целевых функций подсистем, причем компоненты π_k можно интерпретировать как "цены" конфликтных ресурсов, выраженные в единицах глобальной критериальной функции. Информационный обмен между координатором и подсистемами аналогичен представленному на рис.15.3, координатор (Центр) посылает в подсистемы значения $\pi_{k\nu}$ $(\pi^{т}_{k\nu} = \pi^{т}_{k}A^{c}_{\nu})$, а подсистемы посылают в Центр значения $u_{\nu j}$, $g^{м}_{\nu j}$, которые представляют собой множества вариантов планов и значений модифицированной целевой функции, сообщаемых на каждом итерационном цикле согласования координатору

$$g^{м}_{\nu j} = (g_\nu(u_{\nu j}) - (\pi_{k\nu}, u_{\nu j})),\ \nu \in N,\ j = 1,\dots,p_\nu.$$

p_ν - количество вариантов планов, сообщаемых подсистемой координатору на итерационном цикле;

$\pi_{k\nu}$ - n-ый подвектор вектора π_k двойственных оценок условий (15.53); $\pi_k = |\pi_{k1},\dots,\pi_{k\nu},\dots,\pi_{kn}|$.

Отметим, что нахождение оптимальных координирующих сигналов $\pi_{k\nu}$ эквивалентно решению исходной задачи планирования.

Оптимизационная задача Центра (координирующая задача) имеет вид (15.52)-(15.55) при условии, что $S_\nu = p_\nu$. Координатор сообщает подсистемам координирующие сигналы $\pi_{k\nu}$, которые представляет собой предварительное распределение "цен" в единицах глобальной целевой функции на использование технических средств управления. Если такой информации у координатора нет, то $\pi_{k\nu}$ представляют собой вектора соответствующий размерности, все компоненты которого равны 0.

Подсистемы на основе моделей, отражающих особенности управления КА, планируют операции и находят оптимальный относительно координирующего сигнала $\pi_{k\nu}$ план $u^{*}_{\nu}(\pi_{k\nu})$. Для повышения скорости сходимости процесса координации планов подсистем целесообразно наряду с оптимальным планом выполнения операций управления КА находить некоторое множество подоптимальных планов. Тогда в подсистемах вырабатываются множества вариантов планов $u_{\nu j}$ и соответствующих им целевых функций $g^{м}_{\nu j}$, где $g^{м}_{\nu j} = (g_\nu(u_{\nu j}) - (\pi_{k\nu}, u_{\nu j}))$. Значения $g^{м}_{\nu j}$, $u_{\nu j}$ сообщаются координатору.

На основе полученных из подсистем данных $(g^{м}_{\nu j}, u_{\nu j})$, $\nu \in N$, $j=1,...,p_\nu$ координатор строит свою задачу в виде (15.52)-(15.55). Для этого необходимо проанализировать конфликты между планами $u_{\nu j}$, $\nu \in N$, $j=1,...,p_\nu$, и формализовать их в виде векторов конфликтов $\alpha_{\nu j} = A^c_\nu\, u_{\nu j}$.

Процедура координации планов КА идейно схожа с координацией Данцига-Вулфа. Действительно, одинаково используется механизм "цен", как координирующих воздействий, аналогично проводится модификации целевых функций подсистем. Вместе с тем имеется ряд существенных отличий:

- целочисленность получаемых решений и, как следствие этого, то, что глобальный план системы в целом получается как объединение планов подсистем в отличие от алгоритма Данцига-Вулфа, где глобальный план - свертка планов подсистем с некоторыми коэффициентами, определяемыми в результате решения координирующей задачи;
- возможность использования в качестве моделей планирования в подсистемах таких моделей, структура которых отлична от вида задач линейного программирования.

Полученный в результате решения рассмотренной задачи оптимальный план учитывает особенности управления КА, определяемые спецификой выполнения целевых задач, и конфликты, возникающие при управлении различными объектами, которые имеют локальный по времени характер. Вместе с тем существует ряд условий функционирования технических средств управления, которые проявляются либо на всем интервале планирования в целом, либо на временных интервалах, соизмеримых с ним. Такие ограничения целесообразно учитывать при решении дополнительной задачи, которое проводится на основе использования идей релаксации (см.15.3). Особенность условий, описываемых такой задачей, заключается в том, что в реальных ситуациях управления они, как правило, выполняются. Кроме того, на практике ряд ограничений носит нестрогий, рекомендательный характер, и в случаях, если других возможностей выполнения операций управления, связанных с целевым назначением КА, нет, они могут не приниматься во внимание. Таким образом, целесообразно производить релаксацию ограничений задачи (временное отбрасывание части условий) и решать релаксированную адачу с последующим учетом отброшенных условий.

Контрольные вопросы

1). По каким причинам необходимо координировать решения в сложной иерархической системе ?

2). Какими способами и за счет чего может осуществляться координация решений ?

3). В чем заключается смысл координирующих сигналов в алгоритмах с модификацией целевой функции ?

4) В чем состоит смысл координирующих сигналов в алгоритмах с модификацией пространства альтернатив ?

5). Кто является инициатором обмена в алгоритме Данцига-Вулфа и какую информацию он сообщает ?

6). При каких условиях заканчивается итерационный процесс координации в соответствии с алгоритмом Данцига-Вулфа ?

7). Каким образом формируется окончательное решение в алгоритме Данцига-Вулфа ?

8). Кто является инициатором обмена в алгоритме Корнаи-Липтака и какую информацию он сообщает ?

9). При каких условиях заканчивается итерационный процесс координации в соответствии с алгоритмом Корнаи-Липтака ?

10). Какие задачи решает Центр и подсистемы в координационном алгоритме Корнаи-Липтака ?

11). В чем заключается практический смысл релаксации в задачах координационного планирования ?

12). По каким причинам необходимо координировать решения в АСУ КА ?

13). Каким образом можно формализовать конфликты при управлении операциями проводимыми КА ?

14). В чем состоит специфика решения координирующей задачи в задачах планирования операций в АСУ КА ?

15). Какие отличия в решении задач подсистем в алгоритме координационного планирования операций КА по сравнению с алгоритмом Данцига-Вулфа ?

16. ПРИМЕРЫ ЗАДАЧ ПРИНЯТИЯ РЕШЕНИЙ В СЛОЖНЫХ СИСТЕМАХ

16.1. Принятие решений в условиях неопределенного воздействия внешней среды

В условиях воздействия среды обобщенная математическая модель ситуации принятия решения (1.2) в составе множеств, образующих множество допустимых альтернатив, включает множество Ω, описывающее состояния среды, т.е.

$$\Delta = \Delta_1 \times \Delta_2 \times \cdots \times \Delta_s \times \Omega \, .$$

Тогда множество допустимых альтернатив, а соответственно и целевая функция $f: \Delta_1 \times \Delta_2 \times \cdots \times \Delta_s \times \Omega \to R^1$, зависят от состояний среды

$$(\Delta(\omega), f(\omega)), \ \ \omega \in \Omega \, .$$

Таким образом, и сами альтернативы и оценка качества этих альтернатив зависят от степени знания среды, и, если состояния среды ω неизвестны, то тогда множество допустимых решений $\Delta(\omega)$ и целевая функция $f(\omega)$ неопределены. В этих условиях важно выяснить, что известно о среде. И в зависимости от этого строить ту или иную модель принятия решения. Различают три вида среды с неопределенностью (см.13.1): 1) *стохастическая* среда - это среда, состояния которой повторяются в массовых явлениях (статистически устойчивы), следовательно, частоту появления состояний можно описывать количественно (задавать вероятность состояний); 2) *целенаправленная* среда - известны цели, в соответствии с которыми среда выбирает свои состояния, и в зависимости от того, каковы эти цели можно строить рациональное поведение; 3) *неизвестная* среда - это среда относительно которой отсутствуют объективные данные о возможных состояниях или целях, которые она преследует, а имеются лишь предположения лица, принимающего решение (или экспертов).

Рассмотрим примеры задач принятия решений.

16.1.1. Стохастическая среда.

Задача 16.1.1. Результаты анализа состояния космического пространства в ближней операционной зоне необходимо передавать для их последующего анализа в Главный центр. При этом может использоваться четыре независимых направления связи x_1, x_2, x_3, x_4, эффективность использования которых зависит от времени суток и состояния ионосферы. Анализ факторов, влияющих на качество передачи информации, позволяет выявить 6 различных состояний среды $w_1, w_2, w_3, w_4, w_5, w_6$, причем многолетние наблюдения позволяют заключить, что вероятность этих состояний задается как $p = (0.05\ 0.1\ 0.35\ 0.1\ 0.15\ 0.25)$. Необходимо выявить наилучшее в этих условиях направление связи, если качество передачи информации при различных состояниях среды задается таблицей

	w_1	w_2	w_3	w_4	w_5	w_6
x_1	300	700	400	500	800	600
x_2	400	600	500	600	900	400
x_3	500	700	600	700	300	700
x_4	800	600	300	500	400	700

Анализ ситуации показывает, что среда не влияет на имеющиеся альтернативы выбора, а влияет на оценку качества альтернатив. Т.е. имеет место ситуация $(\Delta=\{x_1,x_2,x_3,x_4\},\ f(x,\omega))$, $\omega\in\Omega = \{\omega_1,\omega_2,\omega_3,\omega_4,\omega_5,\omega_6\}$, при которой $f(x,\omega)$, является случайной функцией. Тогда можно произвести детерминизацию задачи и найти решение доставляющее максимум математическому ожиданию целевой функции. В этом случае (см.(13.2))

$$x^* = \arg\max_{x\in\Delta} f^m(x), \qquad f^m(x) = M\, f(x, \omega)\ .$$

Или

$$f^m(x_i) = \sum_{j=1}^{6} f(x_i, \omega_j)\, p_j\ , \quad i = 1,\ldots,4.$$

Значения $f(x_i,\omega_j)$, $i=1,...,4$, $j=1,...,6$ заданы в таблице. Производя необходимые расчеты, получаем значения функции математического ожидания качества передачи информации $f^m(x) = (545, 550, 595, 490)$, откуда оптимальное решение $x^* = x_3$, т.к. $f^m(x_3) = 595$.

16.1.2. Целенаправленная среда.

Задача 16.1.2.

Для анализа состояния сил и средств противника в районе ведения боевых действий необходимо осуществлять сбор информации и передачу формализованных сообщений в Центр, при этом может использоваться четыре независимых направления связи A_1,A_2,A_3,A_4. Противник обладает аппаратурой и возможностями создания пяти видов помех B_1,B_2,B_3,B_4, B_5. Качество передачи информации по тому или иному направлению связи в условиях создания помех характеризуется матрицей F

	B_1	B_2	B_3	B_4	B_5
A_1	0.3	0.7	0.4	0.5	0.8
A_2	0.4	0.6	0.5	0.6	0.9
A_3	0.5	0.7	0.6	0.7	0.9
A_4	0.8	0.6	0.8	0.5	0.4

Необходимо найти рациональный вариант передачи сообщений, гарантирующий максимальное качество информации.

Анализ задачи показывает, что неизвестен вариант создания помех, который выберет противник, но он будет выбирать такой вариант, чтобы качество передачи информации было наименьшим, В этих условиях ситуация может быть формально представлена в виде матричной игры с платежной матрицей F. Тогда, если информация передается однократно, то следует выбирать вариант, соответствующий гарантирующей стратегии (см.13.3.2). Это вариант, на котором достигается нижняя цена игры (см.(13.14))

$$i^* = \arg\max_i F_i^{(-)} = \arg\max_i \min_j f_{ij}.$$

В рассматриваемой задаче $F_i^{(-)} = (0.3\ 0.4\ 0.5\ 0.4)$, следовательно, гарантирующая стратегия – третья $i^*=3$, а нижняя цена игры - гарантированное качество передачи информации, $F^{(-)}= 0.5$.

В тоже время верхняя цена игры, характеризующая максимально возможное качество передачи информации при рациональном выборе вида помех противником (см.(13.15)), равна

$$F^{(+)} = \arg \min_{i} \max_{j} f_{ij} = \min \{0.8\ 0.7\ 0.8\ 0.7\ 0.9\} = 0.7.$$

Следовательно, игра не разрешима в чистых стратегиях, и третья стратегия первого игрока не является оптимальной. Тогда существует возможность повысить качество передачи информации, комбинируя варианты направлений связи. Такая оптимальная комбинация, в соответствии с основной теоремой теории игр всегда существует, и описывается оптимальной смешанной стратегией p^*.

Найдем такую смешанную стратегию, для чего вначале произведем преобразование платежной матрицы (умножим ее элементы на 10 и вычтем из результата 3). Анализируя стратегии первого игрока, можно сделать вывод, что первая и вторая стратегии доминируются третьей. Тогда

$$F = \begin{array}{c} \\ A_3 \\ A_4 \end{array} \begin{array}{c} \begin{array}{ccccc} B_1 & B_2 & B_3 & B_4 & B_5 \end{array} \\ \left\| \begin{array}{ccccc} 2 & 4 & 3 & 4 & 6 \\ 5 & 3 & 5 & 2 & 1 \end{array} \right\| \end{array}$$

Анализируя стратегии второго игрока, можно сделать вывод, что третья стратегия доминируется первой, а вторая стратегия доминируется четвертой. Тогда

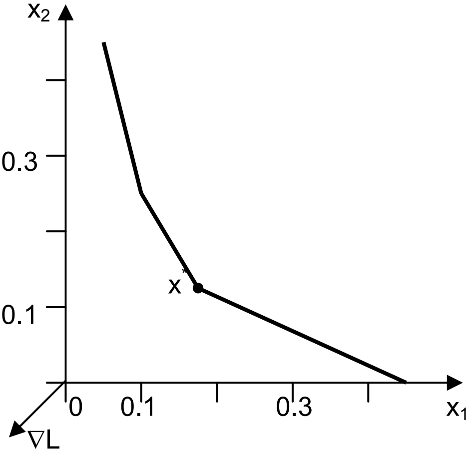
$$F = \begin{array}{c} \\ A_3 \\ A_4 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_4 & B_5 \end{array} \\ \left\| \begin{array}{ccc} 2 & 4 & 6 \\ 5 & 2 & 1 \end{array} \right\| \end{array}$$

Больше доминируемых стратегий первого и второго игрока нет.

Найдем оптимальную смешанную стратегию первого игрока на основе решения задачи линейного программирования следующего вида

$$L = x_1 + x_2 \to \min$$
$$2x_1 + 5x_2 \geq 1,$$
$$4x_1 + 2x_2 \geq 1,$$
$$6x_1 + 1x_2 \geq 1,$$
$$x_i \geq 0,\ i = 1,2.$$

Решим данную задачу графо-аналитическим способом.



В результате решения задачи получим
$L = 0.3125$; $x_1 = 0.1875$; $x_2 = 0.125$. Тогда $C = 3.2$; $p = (0.6\ 0.4)$.

Решим данную задачу с использованием итерационного алгоритма Брауна-Роббинсон (см. 13.3.3), для чего проделаем 10 итераций.

$$F = \begin{array}{c} \\ A_3 \\ A_4 \end{array} \begin{array}{c} \begin{array}{ccc} B_1 & B_4 & B_5 \end{array} \\ \left\| \begin{array}{ccc} 2 & 4 & 6 \\ 5 & 2 & 1 \end{array} \right\| \end{array}$$

к	i	Z_{11}	Z_{12}	Z_{13}	j	Z_{21}	Z_{22}	C_1	C_2	C
1	1	***2***	4	6	1	2	***5***	2.0	5.0	3.5
2	2	7	***6***	7	2	6	***7***	3.0	3.5	3.3
3	2	12	***8***	8	2	***10***	9	2.7	3.3	3.0
4	1	14	***12***	14	2	***14***	11	3.0	3.5	3.3
5	1	16	***16***	20	2	***18***	13	3.2	3.6	3.4
6	1	***18***	20	26	1	***20***	18	3.0	3.3	3.2
7	1	***20***	24	32	1	22	***23***	2.9	3.3	3.1
8	2	***25***	26	33	1	24	***28***	3.1	3.5	3.3
9	2	30	***28***	34	2	28	***30***	3.1	3.3	3.2
10	2	35	***30***	35	2	32	***32***	3.0	3.2	3.1

Анализируя полученные результаты, делаем заключение, что цена игры C=3.1; смешанные стратегии игроков p = (0.5 0.5); q = (0.4 0.6). Данные результаты отличаются от предельных значений, полученных на основе линейного программирования, что объясняется небольшим количеством проведенных итераций. После 100 итерац\ий, проведенных по методу Брауна-Роббинсон получим, что цена игры C = 3.2; смешанные стратегии игроков p = (0.6 0.4); q = (0.4 0.6), что полностью соответствует результатам, полученным на основе решения задачи линейного программирования.

Произведем анализ полученных результатов в терминах исходной постановки задачи:

- для передачи сообщений целесообразно комбинировать использование третьего и четвертого направления связи в соотношении 0.6 и 0.4.

- противнику целесообразно использовать первый и четвертый виды помех в соотношении 0.4 и 0.6.

- качество передачи сообщений определяется ценой игры и составляет 0.62 (для вычисления цены исходной игры необходимо произвести обратные преобразования - добавить к цене игры 3 и разделить на 10).

16.1.3. Неизвестная среда.

Задача 16.1.3. Ведутся боевые действия в горах. Необходимо в кратчайшие сроки вернуть занятую противником высоту, имеющую важное тактическое значение. Погодные условия неустойчивы и существенно влияют на выбранный вариант боя, для проведения которого привлекаются наземные и воздушные силы и средства. Проведенный анализ позволил выявить 4 варианта, которые могут привести к успеху при соответствующем состоянии погодных условий R_i , i = 1,..,5, и различаются привлекаемым составом сил и средств. Матрица F характеризует эффективность выполнения боевой задачи при различных вариантах ведения боя и различных погодных условиях.

$$
F = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{array}
\begin{array}{c} \begin{array}{ccccc} R_1 & R_2 & R_3 & R_4 & R_5 \end{array} \\ \left\| \begin{array}{ccccc} 0.9 & 0.5 & 0.3 & 0.2 & 0.2 \\ 0.3 & 0.6 & 0.9 & 0.7 & 0.3 \\ 0.4 & 0.8 & 0.5 & 0.7 & 0.6 \\ 0.5 & 0.4 & 0.7 & 0.6 & 0.6 \end{array} \right\| \end{array}
$$

Если предполагать, что состояния среды равновероятны, то исходная ситуация принятия решения в условиях неизвестнсти сводится к принятию решений в стохастической среде. Тогда использование критерия Лапласа позволяет оценить математическое ожидание эффективности решения боевой задачи, и в этих условиях наиболее предпочтителен вариант 3 (математическое ожидание ээфективности операции 0.6).

Если предполагать, что погодные условия будут наиболее неблагоприятны для проведения боя, то использование критерия Вальда позволяет выделить две стратегии (3 и 4), на которых достигается гарантированное значение эфективности выполнения боевой задачи (максимин) равное 0.4.

По критерию максимального оптимизма наиболее предпочтительны стратегии 1 и 2; значение функции выигрыша равно 0.9.

По критерию Гурвица стратегия 3 оптимальна при значениях меры оптимизма g в пределах 0.0 - 0.5; стратегия 2 оптимальна в пределах 0.5 -1.0 (в точке g = 1.0 стратегии 1 и 2 равноценны). Стратегия 4 по данному критерию доминируется стратегией 3.

Для анализа стратегий по критерию Сэвиджа построим матрицу риска Z. Элементы данной матрицы получаются из элементов платежной матрицы F путем вычитания значения элемента f_{ij} каждого j–го столбца из значения, которое является максимальным для данного столбца (см. 13.4.1).

$$Z = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \begin{array}{c} \begin{array}{ccccc} R_1 & R_2 & R_3 & R_4 & R_5 \end{array} \\ \begin{Vmatrix} 0.0 & 0.3 & 0.6 & 0.5 & 0.4 \\ 0.6 & 0.2 & 0.0 & 0.0 & 0.3 \\ 0.5 & 0.0 & 0.4 & 0.0 & 0.0 \\ 0.4 & 0.4 & 0.2 & 0.1 & 0.0 \end{Vmatrix} \end{array}$$

По критерию Сэвиджа оптимальной является 4-ый вариант проведения боя, при этом максимальный риск составляет 0.4; для стратегии 3 максимальный риск 0.5; для 1-ой и 2-ой стратегий 0.6.

Анализ результатов проведенного исследования позволяет рекомендовать использование 3-го варианта решения боевой задачи. Это обусловлено тем, что

- данный вариант наиболее предпочтителен при равных вероятностях состояний среды (оценка математического ожидания эффективности 0.6);

- соответствующая стратегия является гарантирующей (значение эффективности решения боевой задачи 0.4);
- по критерию Гурвица данная стратегия доминирует другие в диапазоне пессимистического ожидания благоприятного состояния погоды g=0,...,0.5 (особый оптимизм при анализе возможных вариантов боевых действий не уместен);
- по критерию Сэвиджа 3-я стратегия уступает 4-ой (величина разности критериальных функций равна 0.1), но при трех вариантах состояния погодных условий (более, чем в 50%) использование данного варианта боя просто оптимально, т.е. риска нет.

Альтернативным вариантом является использование 4-го способа ведения боевых действий. Окончательное решение принимает командир, ответственный за выполнение боевой задачи.

16.2. Принятие решений в условиях критериальной неопределенности

Специфический вид неопределенности - критериальная неопределенность вызвана тем, что лицо, принимающее решение, предъявляет несколько различных качественных требований, которым должно удовлетворять решение. Каждое требование описывается некоторой целевой функцией, и тогда говорят, что критерий оптимальности становится векторным. Под решением задач векторной оптимизации будем понимать здесь выбор оптимального в некотором смысле решения на основе нескольких целевых функций. С точки зрения традиционной математики (математических методов оптимизации) такие задачи являются некорректными, и регуляризация таких задач достигается путем введения некоторого решающего правила, формально представляющего собой либо некоторый оператор, позволяющий сформировать результирующую целевую функцию, либо некоторый алгоритм, позволяющий выделить оптимальное решение. Естественным требованием для всех таких решающих правил является требование выделения на их основе недоминируемых решений (решений, лежащих заведомо в области Парето).

Итак, традиционная для принятия решений схема (концептуальная модель, формальная модель, алгоритмическая модель, программная модель, проведение вычислений) дополняется процедурами выбора решающего правила, которые проводятся на основе тесного взаимодействия с лицом, принимающим решение и несущим за него ответственность. Акт выбора решающего правила обычно существенно неформальный, поэтому по результатам анализа найденного "оптимального" решения задачи векторной оптимизации может возникнуть необходимость некоторого изменения (модификации) решающего правила.

Рассмотрим следующую задачу.

Задача 16.2.1. Для оперативного восполнения орбитальной группировки разведывательных КА необходимо в кратчайшие сроки организовать выпуск новых образцов специальной аппаратуры. Основу аппаратуры составляют комплектующие изделия двух типов A_1 и A_2, которые с учетом перспектив развития военно-политической обстановки целесообразно выпускать в максимальных объемах в рамках ограничений имеющихся на предприятии ресурсов B_1, B_2, B_3. При этом технические возможности предприятия не позволяют выпустить комплектующих A_1 более 400 единиц, а комплектующих A_2 более 300 единиц. С другой стороны планировать выпуск комплектующих изделий в объемах менее 80 и 40 единиц соответственно не имеет смысла.

Количество ресурсов B_1, B_2, B_3, которые могут быть выделены на предприятии для производства аппаратуры нового вида, и потребности этих ресурсов для производства единицы комплектующих изделий A_1, A_2 представлены как

	A_1	A_2
$B_1 = 1760$	4	2
$B_2 = 1080$	2	2
$B_3 = 1320$	1	4

Необходимо спланировать выпуск комплектующих изделий в максимальных количествах с учетом имеющихся ограничений.

Решение (план выпуска) в такой задаче можно описать вектором $x = (x_1, x_2)$, где x_1 - объем выпуска комплектующих первого типа; x_2 - объем выпуска комплектующих второго типа. Тогда $80 \le x_1 \le 400$; $40 \le x_2 \le 300$. С учетом потребностей выпуска продукции в ресурсах ограничения можно представить в виде

$$\begin{aligned} 4x_1 + 2x_2 &\le 1760, & 2x_1 + x_2 &\le 880,\\ 2x_1 + 2x_2 &\le 1060, \quad \text{или} & x_1 + x_2 &\le 540,\\ x_1 + 4x_2 &\le 1320, & x_1 + 4x_2 &\le 1320. \end{aligned}$$

В качестве целевых функций необходимо использовать две функции вида

$f_1 = x_1 \rightarrow \max,$
$f_2 = x_2 \rightarrow \max.$

Тогда поставленную задачу формально можно представить в виде:

$$f_1 = x_1 \rightarrow \max,$$
$$f_2 = x_2 \rightarrow \max. \qquad (16.1)$$
$$2x_1 + x_2 \le 880,$$
$$x_1 + x_2 \le 540, \qquad (16.2)$$
$$x_1 + 4x_2 \le 1320,$$
$$80 \le x_1 \le 400, \quad 40 \le x_2 \le 300. \qquad (16.3)$$

Данная задача относится к классу задач линейного программирования с векторной целевой функцией. Т.к. в постановке задачи присутствуют только две переменные, будем решать ее графическим способом. На рис. 16.1 представлена допустимая область изменения неизвестных переменных.

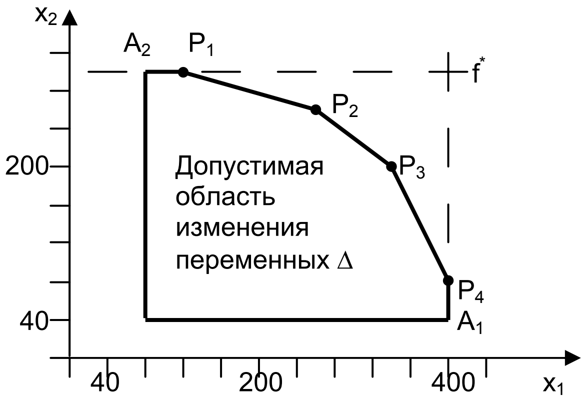


Рис. 16.1.

Решения, оптимальные по 1-му показателю, лежат на отрезке $[A_1,P_4]$, решения, оптимальные по 2-му показателю, лежат на отрезке $[A_2,P_1]$; f^* - идеальная точка, в которой доставляются оптимальные значения обоих показателей, не принадлежит допустимой области изменения переменных. Множество недоминируемых решений (область Парето) лежит на гранях многогранника решений (в данном случае это отрезки прямых), соединяющих недоминируемые вершины многогранника P_1,P_2,P_3,P_4. Исходя из ограничений задачи, недоминируемые вершины имеют следующие координаты: $P_1(120,300)$; $P_2(280,260)$; $P_3(340,200)$; $P_4(400,80)$.

Для нахождения конкретного решения необходимо доопределить данную неопределенную ситуацию на основе взаимодействия с лицом, принимающим решение (ЛПР).

<u>Операторные решающие правила.</u>

<u>Задача 16.2.2.</u> В условиях задачи 16.2.1. на основе взаимодействия с ЛПР принято, что оптимальным следует считать решение, доставляющее минимум сумме относительных отклонений от оптимальных значений показателей.

Относительное отклонение $d_i(x)$ значения i-ой целевой функции от оптимального ее значения для решения x можно вычислить как (см. ф. (14.10))

$$d_i(x) = (f^*_i - f_i(x)) / f^*_i,$$

где f^*_i - оптимальное значение по i-ой целевой функции.

Тогда значение относительного отклонения по 1-ой целевой функции:

$$d_1(x) = (400 - x_1) / 400 = 1 - x_1 / 400,$$

по 2-ой целевой функции:

$$d_2(x) = (300 - x_2) / 300 = 1 - x_2 / 300.$$

С учетом этого критерий оптимальности в данном примере можно записать в виде:

$$d_1(x) + d_2(x) = (1 - x_1/400) + (1 - x_2/300) \rightarrow \min.$$

Учитывая, что из выражения для показателя можно вычитать (или добавлять) константу, а также домножать показатель на положительное число, последнее выражение можно переписать

$$3x_1 + 4x_2 \rightarrow \max.$$

Производя поиск оптимального решения по данной целевой функции на множестве допустимых решений Δ, которое описывается ограничениями (16.2), (16.3) (см. рис.16.1), получаем, что оптимальное решение находится в точке $P_2(280,260)$. В данной точке сумма относительных отклонений составляет 0.43.

<u>Задача 16.2.3.</u> В условиях задачи 16.2.1. на основе взаимодействия с ЛПР принято, что оптимальным следует считать решение, минимизирующее верхнюю грань относительных отклонений от оптимальных значений 1-ой и 2-ой целевой функции.

Из предыдущей задачи следует, что относительное отклонение по 1-ой целевой функции:

$$d_1(x) = 1 - x_1/400,$$

по 2-ой целевой функции:

$d_2(x) = 1 - x_2/300.$

Тогда задачу можно записать как (см. (14.12)):

$$\begin{aligned} & w \to \min, \\ & 1 - x_1/400 \le w, \\ & 1 - x_2/300 \le w, \\ & \Delta,\ w \ge 0. \end{aligned}$$

Здесь Δ - область изменения переменных, определяемая ограничениями (16.2), (16.3). Иначе последнюю задачу можно представить в виде

$$\begin{aligned} & w \to \min, \\ & x_1 + 400\, w \ge 400, \\ & x_2 + 300\, w \ge 300, \\ & \Delta,\ w \ge 0. \end{aligned}$$

Решая данную задачу с использованием симплекс-метода, получаем оптимальное решение $x_1 = 309$; $x_2 = 231$. При этом максимальные относительные отклонения значений целевых функций от оптимальных значений в этой точке не превышают 0.23. Данное решение соответствует точке W(309,231) на грани (P_2,P_3).

Отметим, что верхняя грань относительного отклонения значений целевых функций от оптимальных значений, вычисленная для оптимального решения предыдущей задачи (точка P_2(280,260)), составляет 0.3. В тоже время сумма относительных отклонений, вычисленная в точке W(309,231), составляет 0.46 (для точки P_2(280,260) - оптимального решения предыдущей задачи, она составляет 0.43).

<u>Задача 16.2.4.</u> В условиях задачи 16.2.1. требуется найти решение, соответствующее максимальному приближению к идеальной точке f^*. Из рис.16.1 очевидно, что такая точка лежит на ребре (P_2,P_3). Тогда точка F, максимально приближенная к идеальной, определяется как

$$F = \alpha P_2 + (1 - \alpha)\, P_3,$$

где α - параметр, $\alpha \in [0,1]$. Значение параметра α выбирается из условия минимума расстояния (в метрике Евклида) между точками F и f^*; обозначим такое расстояние как $r(F,f^*)$.

Точка F имеет координаты F(x_1,x_2), точки P_2 и P_3 имеют координаты P_2(280,260); P_3(340,200). Тогда

$$x_1 = \alpha\, 280 + (1 - \alpha)\, 340 = 60\, \alpha + 340;$$

$x_2 = \alpha\ 260 + (1 - \alpha)\ 200 = 60\ \alpha + 200.$

Квадрат расстояния

$$r^2(F,f^*) = (400 - x)^2 + (300 - y)^2 =$$
$$= (60 - 60\ \alpha)^2 + (100 - 60\ \alpha)^2 \rightarrow \min.$$

Из необходимого условия существования экстремума функции $r^2(F,f^*)$ (равенство 0 первой производной) находится значение параметра α, оно равно 1/3. Тогда точка F, соответствующая минимуму расстояния $r(F,f^*)$, имеет координаты F(320, 220). Сравнивая данное решение с решением, полученным в предыдущей задаче, можно заключить, что эти решения достаточно близки (хотя расстояние от точки W до идеальной точки на 14% больше, чем от точки F). В то же время последнее решение (решение данной задачи) доставляет большие максимальные относительные отклонения от оптимумов, а именно 0.27 (по сравнению с 0.23 в предыдущей задаче). С другой стороны, в общем случае, нахождение решения, максимально близкого к идеальной точке, требует решения задачи нелинейного программирования, в то время как нахождение решения, минимизирующего максимальное относительное отклонение от оптимальных значений показателей, не выводит из класса решаемых задач, т.к. дополнительные ограничения являются линейными.

Задача 16.2.5. В условиях задачи 16.2.1. на рассматриваемом временном интервале в целях скорейшего выпуска аппаратуры изготовление комплектующих изделий 1-го типа оценивается в 1.5 раза важнее, чем изделий 2-го типа. Это соответствует получению экспертной информации об относительной важности показателей в виде ряда приоритетов $(f_1\ f_2)$; $\lambda = (1\ 1.5)$.

Построим результирующую целевую функцию в форме (14.7) для этого необходимо определить коэффициенты относительной важности.

$\alpha_1 = 1.5\ \alpha_2,$

$\alpha_1 + \alpha_2 = 1.$

Откуда $\alpha_1 = 0.6$; $\alpha_2 = 0.4$. Тогда результирующая целевая функция имеет вид

$$f_{рез} = 3x_1 + 2x_2 \rightarrow \max_{\Delta}.$$

Оптимальным решением (см. рис. 16.1) является точка P_3 (340,200), которая обеспечивает выпуск комплектующих изделий 1-го типа в 1.7 раза больше, чем изделий 2-го типа.

<u>Задача 16.2.6.</u> В условиях задачи 16.2.1. при скорейшем выпуске новой аппаратуры испытывается существенный недостаток комплектующих изделий 1-го типа. В этих условиях оптимизацию целесообразно проводить по последовательно применяемым критериям $f=(f_1,f_2)$, где

$f_1= x_1 \to \max$

$f_2= x_2 \to \max.$

Тогда (см. (14.20))

$\Delta^1 = \{(x_1,x_2)\in\Delta \mid x_1 = 400;\ 40\le x_2\le 80 \}$, т.е. это участок (P_4,A_1) - см. рис. 16.1 или рис. 16.2.

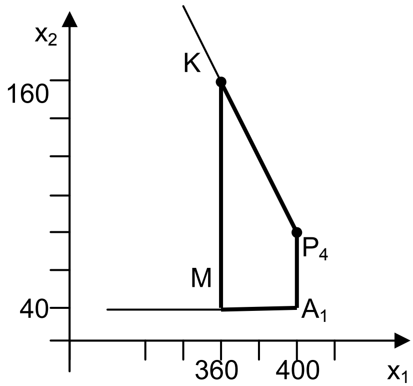


Рис.16.2.

Использование критерия f_2 позволяет выделить единственную точку P_4 в качестве оптимальной, $P_4(400,80)$.

<u>Задача 16.2.7.</u> В условиях задачи 16.2.6 (см. рис.16.2) оптимизация идет по последовательно применяемым критериям. При этом допускается снижение значений f_1 на 10%. Требуется найти оптимальное решение.

Максимальное значение $\max f_1 = 400$, тогда абсолютное значение уступки $\varepsilon^1 = 40$. В этом случае с учетом (14.20)

$\Delta^1 = \{ (x_1,x_2)\in\Delta \mid x \ge 360 \}$.

На рис. 16.2 это многогранник, ограниченный точками A_1, P_4, K, M. Максимизируя на Δ^1 значения целевой функции $f_2=x_2$, получаем оптимальное решение в точке K(360,160). Сравнивая это решение с решением предыдущей задачи, можно заключить, что значение 1-ой целевой функции уменьшилось на 10%, а значение 2-ой целевой функции увеличилось в 2 раза.

Сделанные замечания по особенностям решения задач векторной оптимизации и приведенные примеры показывают, что разнообразные формы задания решающих правил и способы учета информации об относительной важности критериев представляют собой достаточно эффективный инструмент решения задач, позволяющий лицу, принимающему решение, адекватно учитывать особенности ситуации оптимального выбора.

Лингвистическое решающее правило.

Задача 16.2.7. В результате предварительного анализа было разработано 10 вариантов оперативной модернизации производства военного предприятия $x_1, x_2, ..., x_{10}$. Каждый вариант оценивается по трем критериям: f_1 - минимум стоимости переоборудования производства; f_2 - минимум времени, необходимого для модернизации производства; f_3 - максимум эффективности нового производства. Значения соответствующих целевых функций, вычисленные для каждого решения, представлены в таблице

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
f_1	306	246	282	210	278	294	234	222	270	218
f_2	16.0	26.7	21.0	31.9	24.9	18.5	28.4	30.2	23.2	33.3
f_3	3.42	5.12	5.36	4.19	3.84	4.86	4.07	3.82	4.12	4.06

Анализ данных, представленных в таблице, позволяет заключить, что вариант x_5 доминируется вариантом x_9 (x_9 является лучшим по всем показателям), а вариант x_{10} доминируется вариантом x_4. Тогда таблицу можно переписать

	x_1	x_2	x_3	x_4	x_6	x_7	x_8	x_9
f_1	306	246	282	210	294	234	222	270
f_2	16.0	26.7	21.0	31.9	18.5	28.4	30.2	23.2
f_3	3.42	5.12	5.36	4.19	4.86	4.07	3.82	4.12

Рассмотрим особенности принятия решения на основе лингвистического подхода. При этом используется понятие лингвистической переменной (см. 13.4.2), которая описывается математической конструкцией вида $L_p = (I,T,U,Г,M)$, где I - имя лингвистической переменной - "решение"; T - терм -множество значений лингвистической переменной (каждому элементу из терм-множества соответствует некоторое нечеткое множество); U -

универсальное множество объектов - множество недоминируемых вариантов решений $U = \{x_1,x_2,x_3,x_4,x_6,x_7,x_8,x_9\}$; Г - синтаксическое правило (грамматика), порождающее множество элементов терм-множества; М - семантическое правило, ставящее в соответствие значению лингвистической переменной ее смысл, т.е. правило, сопоставляющее некоторому терму нечеткое множество вариантов решений. Принятие решение на основе лингвистической переменной осуществляется по схеме (см. (14.21))

$$S \xrightarrow[1]{\text{ЛПР}} \{M_i,\ i \in G\} \xrightarrow[2]{\text{ЛПР}} L \xrightarrow[3]{} C(L) \xrightarrow[4]{} M_L \xrightarrow[5]{} x^*.$$

Рассмотрим последовательность действий, которые необходимо провести в соответствии с этой схемой.

(1). На основе взаимодействия с ЛПР вводятся нечеткие множества "хороших по i-му показателю решений", i = 1,2,3. Нечеткое множество M_i задается конструкцией (см. 13.4.2) вида $M_i = \{ (x, \mu_i(x))\}$, где $x \in U = \{x_1,x_2,x_3,x_4,x_6,x_7,x_8,x_9\}$. Поскольку универсальное множество конечно, то нечеткое множество M_i можно задавать вектором m_i из восьми компонент, каждая компонента которого характеризует степень принадлежности $\mu_i(x_j)$ соответствующего x_j-го решения к нечеткому множеству "хороших по i-му показателю решений". Расчет степени принадлежности $\mu_i(x)$ будем проводить с использованием (14.22)

$$\mu_i(x) = \frac{f_i(x) - f^l_i(x)}{f_i(x^*) - f^l_i(x)}, \quad i=1,2,3.$$

Здесь $f^l_i(x) = \mathrm{Min}\, f_i(x) - \varepsilon$, $x \in U$. Значение ε (по согласованию с ЛПР) выбирается таким образом, чтобы $\min \mu_i(x) = 0.1$.

Рассмотрим построение нечеткого множества "хороших по 1-му показателю решений". Т.к. необходимо минимизировать значение первой целевой функции, то для того, чтобы использовать приведенные в 14.4 расчетные формулы необходимо свести ее к максимуму. Тогда

	x_1	x_2	x_3	x_4	x_6	x_7	x_8	x_9
f_1:	-306	-246	-282	-210	-294	-234	-222	-270

В этом случае $\mathrm{Min}\, f_1(x) = -306$; $f_1(x^*) = -210$; тогда ε находится из уравнения

$$\mathrm{Min}\, f_1(x) - \mathrm{Min}\, f_1(x) + \varepsilon = 0.1\, (f_1(x^*) - \mathrm{Min}\, f_1(x) + \varepsilon)$$

или $\quad 0.9\,\varepsilon = 0.1\,(f_1(x^*) - \mathrm{Min}\, f_1(x)) = 9.6$

откуда $\quad \varepsilon = 11$ и $f^l_i(x) = -317$.

В этом случае нечеткое множество M_1 задается вектором m_1 = (0.10 0.66 0.32 1.00 0.21 0.78 0.89 0.44). Проводя аналогичные вычисления для 2-го и 3-го критериев, получаем значения степеней принадлежности для нечетких множеств M_2, M_3

m_2 = (1.00 0.39 0.72 0.10 0.86 0.30 0.20 0.59),

m_3 = (0.28 0.89 1.00 0.46 0.77 0.40 0.10 0.42).

Таким образом задан смысл первичных простых термов "хорошее по i-му показателю решение", i = 1,2,3..

(2). По согласованию с ЛПР формируется лингвистическое решающее правило L. Как отмечалось ранее в 14.4, лингвистическое решающее правило, по существу, представляет собой составной терм из терм-множества значений лингвистической переменной. Пусть необходимо найти

L = "очень хорошее по f_3 и не очень плохое по f_1 и f_2 решение".

(3). Произведем синтаксический разбор лингвистического решающего правила. Анализ показывает, простой терм "плохое" является образованным и ему соответствует составной терм "не хорошее". Тогда L будет иметь вид "очень хорошее по f_3 и не очень не хорошее по f_1 и f_2 решение". Результаты синтаксического разбора L можно представить в виде

$$\underbrace{\text{"}\underbrace{\text{очень } \underbrace{\text{хорошее по } f_3}_{M^1_3}}_{M^2_3} \text{ и } \underbrace{\text{не } \underbrace{\text{очень } \underbrace{\text{не } \underbrace{\text{хорошее по } f_1}_{M^1_1}}_{M^2_1}}_{M^3_1}}_{M^4_1} \underbrace{\text{ и } f_2 \text{ решение"}}_{\dots\dots\; M^1_2M^2_2M^3_2M^4_2}}_{M_L}$$

Здесь

M^1_3 - "хорошее по 3-ей целевой функции решение";

M^2_3 - "очень хорошее по 3-ей целевой функции решение";

M^1_1 - "хорошее по 1-ой целевой функции решение";

M^2_1 - "не хорошее по 1-ой целевой функции решение";

M^3_1 - "очень не хорошее по 1-ой целевой функции решение";

M^4_1 - "не очень не хорошее по 1-ой целевой функции решение".

$M^1_2, M^2_2, M^3_2, M^4_2$ аналогичные $M^1_1, M^2_1, M^3_1, M^4_1$ множества для 2-ой целевой функции.

(4) Необходимо найти нечеткое множество, определяющее смысл лингвистического решающего правила M_L. Учитывая интерпретации лингвистических связок и лингвистических неопределенностей в операционных преобразованиях нечетких множеств (см. 13.4.2) M_L можно представить в виде

$$M_L = M^2_3 \cap M^4_1 \cap M^4_2 = CON(M^1_3) \cap \neg M^3_1 \cap \neg M^3_2 =$$
$$= CON(M^1_3) \cap \neg\, CON\ M^2_1 \cap \neg\, CON\ M^2_2 =$$
$$= CON(M^1_3) \cap \neg\, CON(\neg M^1_1) \cap \neg\, CON(\neg M^1_2)\ .$$

Нечеткие множества $M^1_3 = M_3$, $M^1_1 = M_1$, $M^1_2 = M_2$ были заданы на основе взаимодействия с ЛПР в п. (1) в форме векторов m_1, m_2, m_3 (ввиду того, что универсальное множество конечно). Тогда

$$m_1 = m^1_1 = (0.10\ 0.66\ 0.32\ 1.00\ 0.21\ 0.78\ 0.89\ 0.44),$$
$$m_2 = m^1_2 = (1.00\ 0.39\ 0.72\ 0.10\ 0.86\ 0.30\ 0.20\ 0.59),$$
$$m_3 = m^1_3 = (0.28\ 0.89\ 1.00\ 0.46\ 0.77\ 0.40\ 0.10\ 0.42).$$

Учитывая, что "не" соответствует взятию дополнения в универсальном множестве, при котором $\mu(\neg X) = 1 - \mu(X)$, запишем

$$m^2_1 = \neg\, m^1_1 = (0.90\ 0.34\ 0.68\ 0.00\ 0.79\ 0.22\ 0.11\ 0.56),$$
$$m^2_2 = \neg\, m^1_2 = (0.00\ 0.61\ 0.28\ 0.90\ 0.14\ 0.70\ 0.80\ 0.41).$$

Так как лингвистическая неопределенность "очень" соответствует операции концентрации, при которой $\mu(CON\ X) = \mu^2(X)$, запишем

$$m^3_1 = CON\ m^2_1 = (0.81\ 0.12\ 0.46\ 0.00\ 0.62\ 0.05\ 0.01\ 0.31),$$
$$m^3_2 = CON\ m^2_2 = (0.00\ 0.37\ 0.08\ 0.81\ 0.02\ 0.49\ 0.64\ 0.17),$$
$$m^2_3 = CON\ m^1_3 = (0.07\ 0.79\ 1.00\ 0.21\ 0.59\ 0.16\ 0.01\ 0.18).$$

Учитывая, что "не" соответствует взятию дополнения в универсальном множестве, при котором $\mu(\neg X) = 1 - \mu(X)$, запишем

$$m^4_1 = \neg\, m^3_1 = (0.19\ 0.88\ 0.54\ 1.00\ 0.38\ 0.95\ 0.99\ 0.69),$$
$$m^4_2 = \neg\, m^3_2 = (1.00\ 0.63\ 0.92\ 0.19\ 0.98\ 0.51\ 0.36\ 0.83).$$

Связка "и" соответствует пересечению нечетких множеств, при котором $\mu(X \cap Y) = \min\{\mu(X), \mu(Y)\}$, тогда

$$m_L = m^2_3 \cap m^4_1 \cap m^4_2 = (0.07\ 0.63\ 0.54\ 0.19\ 0.38\ 0.16\ 0.01\ 0.18).$$

(5) Выбор наилучшего в целом решения x^* производится по правилу (см. ф.(3.26)) $x^* = \arg\max \mu_L(x),\ x \in U$, тогда

$$x^* = \arg\max \{0.07\ 0.63\ 0.54\ 0.19\ 0.38\ 0.16\ 0.01\ 0.18\} = x_2.$$

Таким образом, второй вариант модернизации производства является наилучшим в смысле предпочтений, высказанных ЛПР.

16.3. Декомпозиционные методы принятия решений в сложных системах

Специализация функций военно-технической системы, неизбежно возникающая с ростом сложности решаемых системой задач, приводит к тому, что существующая система управления разбивается на совокупность подсистем, решающих специализированные задачи. Появление в военно-технической системе таких подсистем приводит к появлению иерархической структуры, при которой каждая подсистема принимает решение в соответствии со своими собственными целями, не тождественными в общем случае целям других подсистем и системы в целом. В этой ситуации возникает задача координации решений подсистем, и одна из подсистем высшего уровня (Центр) наделяется этими функциями.

Одним из первых алгоритмов, обеспечивающих координацию решений подсистем на основе введения координирующих сигналов, был алгоритм Данцига-Вулфа (см. 15.2). Рассмотрим следующую задачу.

<u>Задача 16.3.1.</u> На оборонном предприятии два цеха выпускают четыре вида военной продукции (1-ый цех выпускает продукцию A_1, A_2 по цене C_1=30, C_2=20; 2-ой цех выпускает продукцию A_3, A_4 по цене C_3=20, C_4=30). Для выпуска единицы продукции в первом цехе используются ресурсы R_1, R_2, а во втором цехе ресурсы Q_1, Q_2, Q_3, запасы которых ограничены. Вместе с тем для выпуска единицы продукции необходимы затраты электроэнергии W_1= W_2= W_3= W_4= 0.1 общее потребление которой ограничено величиной W=16. Количество ресурсов, необходимое для выпуска единицы продукции каждого вида, представлено как

	R_1=28	R_2=18		Q_1=21	Q_2=9	Q_3=14
A_1	0.1	0.3	A_3	0.1	0.1	0.2
A_2	0.4	0.1	A_4	0.3	0.1	0.1

Необходимо спланировать выпуск продукции по критерию максимума суммарной стоимости выпускаемой продукции.

План выпуска продукции будем описывать вектором x = (x_1, x_2, x_3, x_4), где x_i - количество продукции i-го вида выпускаемое оборонным предприятием.

Тогда задача планирования выпуска продукции 1-го цеха имеет вид

$$30\,x_1 + 20\,x_2 \to \max$$
$$0.1x_1 + 0.4x_2 \le 28$$
$$0.3x_1 + 0.1x_2 \le 18$$
$$x_1, x_2 \ge .\ 0,$$

или, производя замену $x_1 = 0.1x_1$; $x_2 = 0.1x_2$, получаем

$$\begin{aligned} &300x_1 + 200x_2 \to \max \\ &1x_1 + \quad 4x_2 \le 28 \\ &3x_1 + \quad 1x_2 \le 18 \\ &x_1, x_2 \ge 0. \end{aligned}$$

Аналогично, задача планирования выпуска продукции 2-го цеха имеет вид

$$\begin{aligned} &200x_3 + 300x_4 \to \max \\ &1x_3 + \quad 3x_4 \le 21 \\ &1x_3 + \quad 1x_4 \le 9 \\ &2x_3 + \quad 1x_4 \le 14 \\ &x_3, x_4 \ge 0. \end{aligned}$$

Решая данные задачи тем или иным методом, можно определить оптимальный выпуск продукции каждым цехом. Однако эти решения в совокупности должны удовлетворять ограничениям по использованию электрической энергии

$$0.1x_1 + 0.1x_2 + 0.1x_3 + 0.1x_4 \le 16,$$

или с учетом замены переменных

$$1x_1 + 1x_2 + 1x_3 + 1x_4 \le 16.$$

В целом задача принятия решения (с учетом того, что можно изменить масштаб изменения целевой функции) имеет вид

$$\begin{aligned} &3x_1 + 2x_2 + 2x_3 + 3x_4 \to \max \\ &1x_1 + 1x_2 + 1x_3 + 1x_4 \le 16 \\ &1x_1 + 4x_2 \qquad\qquad\quad\ \le 28 \\ &3x_1 + 1x_2 \qquad\qquad\quad\ \le 18 \\ &\qquad\qquad 1x_3 + 3x_4 \le 21 \\ &\qquad\qquad 1x_3 + 1x_4 \le 9 \\ &\qquad\qquad 2x_3 + 1x_4 \le 14 \\ &\qquad\qquad x_1, x_2 x_3, x_4 \ge 0. \end{aligned}$$

Такая задача является задачей линейного программирования вида

$$\begin{aligned} &f(x) = c^T x \to \max \\ &A\,x \le b \\ &\quad x \ge 0 \end{aligned}$$

с блочно-диагональной матрицей условий A

$$\|3 \quad 2 \quad 2 \quad 3\| \; x \to \max$$

$$A\,x = \begin{Vmatrix} 1 & 1 & 1 & 1 \\ 1 & 4 & 0 & 0 \\ 3 & 1 & 0 & 0 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 \end{Vmatrix} \times \begin{Vmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Vmatrix} \le \begin{Vmatrix} 16 \\ 28 \\ 18 \\ 21 \\ 9 \\ 14 \end{Vmatrix}$$

$$x = \| x_1\; x_2\; x_3\; x_4 \| \ge 0.$$

Проведем ее решение на основе декомпозиционного алгоритма Данцига-Вулфа (см.15.2). В процессе координации Центр сообщает подсистемам координирующий сигнал π_{ki}, на основе которого производится модификация целевых функций подсистем; $f_i(x_i,\pi_{ki}) = (c^T_i - \pi^T_{ki})\, x_i$. Подсистемы решают свои оптимизационные задачи и сообщают координатору оптимальные решения x^*_{ir} (здесь r - номер итерации) и значения целевых функций $f_{ir} = f_i(x^*_{ir})$.

(0). Задается начальный номер итерации r = 1. Центр сообщает подсистемам координирующий сигнал $\pi^T_{ki} = 0^T$, i = 1,2.

<u>Итерация 1</u>.

(1) При координирующем сигнале $\pi^T_{ki} = 0^T$, i=1,2 подсистемы решают свои оптимизационные задачи

Подсистема 1	Подсистема 2
$3x_1 + 2x_2 \to \max$	$2x_3 + 3x_4 \to \max$
$1x_1 + 4x_2 \le 28$	$1x_3 + 3x_4 \le 21$
$3x_1 + 1x_2 \le 18$	$1x_3 + 1x_4 \le 9$
$x_1,x_2 \ge 0.$	$2x_3 + 1x_4 \le 14 \qquad x_3,x_4 \ge 0.$

Произведем решение задач подсистем графо-аналитическим способом.

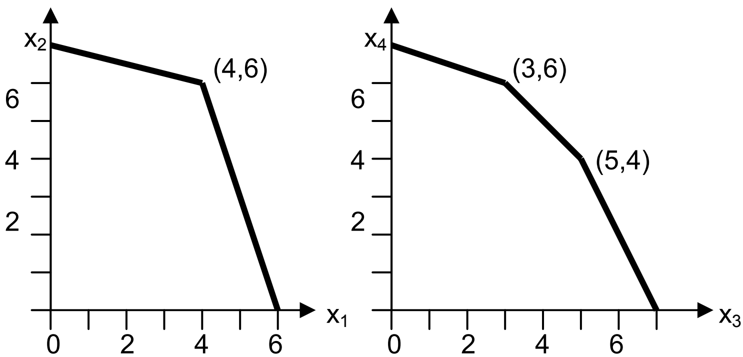


Рис. 16.4.

В результате решения получим

$x^*_{11} = (4, 6)$; $x^*_{21} = (3, 6)$

$f_{11} = c^T_1 x^*_{11} = 24$; $f_{21} = c^T_2 x^*_{21} = 24$.

Значения x^*_{11}, f_{11}; x^*_{21}, f_{21} сообщаются в Центр.

(2) Центр строит координирующую задачу вида (15.23)-(15.26)

$$\sum_{i=1}^{2} \sum_{j=1}^{r} \varphi_{ij} \lambda_{ij} \to \max,$$

$$\sum_{i=1}^{2} \sum_{j=1}^{r} \alpha_{ij} \lambda_{ij} \le b_o ,$$

$$\sum_{i=1}^{2} \sum_{j=1}^{r} \lambda_{ij} = 1, \; i = 1,2,$$

$$\lambda_{ij.} \ge 0, \; i=1,2, \; j=1,\dots, r,$$

Здесь α_{ij} вычисляется по правилу $\alpha_{ij} = A_{oi} x^*_{ij}$. Так как r=1, то

$$\alpha_{11} = \begin{Vmatrix} 1 & 1 \end{Vmatrix} \begin{Vmatrix} 4 \\ 6 \end{Vmatrix} = 10; \quad \alpha_{21} = \begin{Vmatrix} 1 & 1 \end{Vmatrix} \begin{Vmatrix} 3 \\ 6 \end{Vmatrix} = 9.$$

Тогда координирующая задача имеет вид

$$\begin{aligned}
&24\lambda_{11} + 24\lambda_{21} \to \max \\
&10\lambda_{11} + 9\lambda_{21} \le 16 \\
&1\lambda_{11} = 1 \\
&1\lambda_{21} = 1 \\
&\lambda_{11}, \lambda_{21} \ge 0 .
\end{aligned}$$

Такая задача решения не имеет (т.к. при λ_{11}=1, λ_{21}=1. не выполняется ограничение $10\lambda_{11} + 9\lambda_{21} \le 16$). Произведем ее погружение в М-задачу (см. 3.6), причем назначим М = 100:

$$\begin{aligned}
&24\lambda_{11} + 24\lambda_{21} - 100\lambda_4 - 100\lambda_5 \to \max \\
&10\lambda_{11} + 9\lambda_{21} + 1\lambda_3 \le 16 \\
&1\lambda_{11} + 1\lambda_4 = 1 \\
&1\lambda_{21} + 1\lambda_5 = 1 \\
&\lambda_{11}, \lambda_{21}, \lambda_3, \lambda_4, \lambda_5 \ge 0 .
\end{aligned}$$

Произведем решение данной задачи симплекс-методом (см. 3.5). Каждая итерация симплекс-метода сопровождается заполнением таблицы вида

	L	$\pi^т$
J_s	x_s	S^{-1}

Здесь
- L - значение целевой функции;
- $\pi^т$ - транспонированный вектор двойственных оценок;
- J_s - номера базисных переменных;
- x_s - значение базисных переменных;
- S^{-1} - обратная матрица базиса.

Итерация 1;

	-200.00	0.00	-100.00	-100.00
3	16.00	1.00	0.00	0.00
4	1.00	0.00	1.00	0.00
5	1.00	0.00	0.00	1.00

Итерация 2;

	-76.00	0.00	24.00	-100.00
3	6.00	1.00	-10.00	0.00
1	1.00	0.00	1.00	0.00
5	1.00	0.00	0.00	1.00

Итерация 3;

	6.66	13.77	-113.77	-100.00
2	0.66	0.11	-1.11	0.00
1	1.00	0.00	1.00	0.00
5	0.33	-0.11	1.11	1.00

Итерация 4-решение задачи

	10.80	12.40	-100.00	-87.60
2	1.00	0.00	0.00	1.00
1	0.70	0.10	0.00	-0.90
4	0.30	-0.10	1.00	0.90

Итак, решение задачи:

$\lambda_1 = 0.7$; $\lambda_2 = 1.0$; $\lambda_4 = 0.3$.

Здесь $\lambda_1 = \lambda_{11}$; $\lambda_2 = \lambda_{21}$.

(3) Координирующий сигнал $p_к = 12.4$.

Тогда $\pi^т_{к1} = 12.4 \times \| 1\ 1 \| = \| 12.4\ 12.4 \|$;

$\pi^т_{к2} = 12.4 \times \| 1\ 1 \| = \| 12.4\ 12.4 \|$.

Эти сигналы поступают в подсистемы.

(4) Номер итерации $r = r + 1 = 2$.
Производится переход на шаг 1.

Итерация 2.

(1) При координирующих сигналах $\pi^T_{к1} = \| 12.4\ 12.4 \|$; $\pi^T_{к2} = \| 12.4\ 12.4 \|$. Производится модификация целевых функций по правилу $(c^T_i - \pi^T_{ki})\, x_i$, тогда

$f_1(x) = \| 3 - 12.4,\ 2 - 12.4 \|\ x = \| -9.4,\ -10.4 \|\ x \to \max;$

$f_2(x) = \| 2 - 12.4,\ 3 - 12.4 \|\ x = \| -10.4,\ -9.4 \|\ x \to \max;$

Оптимальное решение при таких целевых функциях (см. рис.16.4)

$x^*_{12} = (0, 0);\ f_{12} = 0;$

$x^*_{22} = (0, 0);\ f_{22} = 0.$

Эти данные сообщаются Центру.

(2) Центр строит координирующую задачу. При этом рассчитываются

$$\alpha_{12} = \| 1\quad 1 \| \begin{Vmatrix} 0 \\ 0 \end{Vmatrix} = 0; \qquad \alpha_{22} = \| 1\quad 1 \| \begin{Vmatrix} 0 \\ 0 \end{Vmatrix} = 0.$$

Тогда координирующая задача имеет вид

$$\begin{array}{l} 24\lambda_{11} + 24\lambda_{21} \to \max \\ 10\lambda_{11} + 9\lambda_{21} + 1\lambda_3 \le 16 \\ 1\lambda_{11} + 1\lambda_{12} = 1 \\ 1\lambda_{21} + 1\lambda_{22} = 1 \\ \lambda_{11}, \lambda_{21}, \lambda_{12}, \lambda_{22}, \lambda_3 \ge 0\ . \end{array}$$

Дополнительные переменные λ_4, λ_5 исключены из координирующей задачи, т.к. их роль теперь выполняют переменные λ_{12}, λ_{22}.

Решаем координирующую задачу симплекс-методом.

Итерация 1;

	0.00	0.00	0.00	0.00
3	16.00	1.00	0.00	0.00
4	1.00	0.00	1.00	0.00
5	1.00	0.00	0.00	1.00

Итерация 2;

	24.00	0.00	24.00	0.00
3	6.00	1.00	-10.00	0.00
1	1.00	0.00	1.00	0.00
5	1.00	0.00	0.00	1.00

Итерация 3;

	40.00	2.66	-2.66	0.00
2	0.66	0.11	-1.11	0.00
1	1.00	0.00	1.00	0.00
5	0.33	-0.11	1.11	1.00

Итерация 4 - решение задачи

	40.80	2.40	0.00	2.40
2	1.00	0.00	0.00	1.00
1	0.70	0.10	0.00	-0.90
4	0.30	-0.10	1.00	0.90

Итак, решение задачи:

$\lambda_1 = 0.7$; $\lambda_2 = 1.0$; $\lambda_4 = 0.3$.

Здесь $\lambda_1 = \lambda_{11}$; $\lambda_2 = l_{21}$; $\lambda_4 = \lambda_{12}$.

(3) Координирующий сигнал $\pi_к = 2.4$.

Тогда $\pi^т_{к1} = 2.4 \times \| 1\ 1 \| = \| 2.4\ 2.4 \|$;

$\pi^т_{к2} = 2.4 \times \| 1\ 1 \| = \| 2.4\ 2.4 \|$.

Эти сигналы поступают в подсистемы.

(4) Номер итерации r = r + 1 = 3.

Производится переход на шаг 1.

<u>Итерация</u> 3.

(1) При координирующих сигналах $\pi^т_{к1} = \| 2.4\ 2.4 \|$; $\pi^т_{к2} = \| 2.4\ 2.4 \|$. Производится модификация целевых функций по правилу $(c^т_i - \pi^т_{ki})\, x_i$, тогда

$f_1(x) = \| 3 - 2.4,\ 2 - 2.4 \|\, x = \| 0.6, -0.4 \|\, x \rightarrow \max$;

$f_2(x) = \| 2 - 2.4,\ 3 - 2.4 \|\, x = \| -0.4,\ 0.6 \|\, x \rightarrow \max$;

Оптимальное решение при таких целевых функциях (см. рис.16.4)

$x^*_{13} = (6, 0)$; $f_{13} = 18$;

$x^*_{23} = (0, 7)$; $f_{23} = 21$.

Эти данные сообщаются Центру.

(2) Центр строит координирующую задачу. При этом рассчитываются

$$\alpha_{12} = \| 1\quad 1 \| \begin{Vmatrix} 6 \\ 0 \end{Vmatrix} = 6; \qquad \alpha_{22} = \| 1\quad 1 \| \begin{Vmatrix} 0 \\ 7 \end{Vmatrix} = 7.$$

Тогда координирующая задача имеет вид

$$\begin{array}{lllllll}
24\lambda_{11} + 24\lambda_{21} + 18\lambda_{13} + 21\lambda_{23} & & & & & & \to \max \\
10\lambda_{11} + 9\lambda_{21} + 6\lambda_{13} + 7\lambda_{23} + 1\lambda_3 & & & & & & \le 16 \\
1\lambda_{11} + 1\lambda_{13} + 1\lambda_{12} & & & & & & = 1 \\
1\lambda_{21} + 1\lambda_{23} + 1\lambda_{22} & & & & & & = 1 \\
\lambda_{11}, \lambda_{21}, \lambda_{12}, \lambda_{22}, \lambda_{13}, \lambda_{23}, \lambda_3 \ge 0 .
\end{array}$$

Решаем данную координирующую задачу симплекс-методом.

Итерация 1;

	0.00	0.00	0.00	0.00
5	16.00	1.00	0.00	0.00
6	1.00	0.00	1.00	0.00
7	1.00	0.00	0.00	1.00

Итерация 2;

	24.00	0.00	24.00	0.00
5	6.00	1.00	-10.00	0.00
1	1.00	0.00	1.00	0.00
7	1.00	0.00	0.00	1.00

Итерация 3;

	40.00	2.66	-2.66	0.00
2	0.66	0.11	-1.11	0.00
1	1.00	0.00	1.00	0.00
7	0.33	-0.11	1.11	1.00

Итерация 4 - решение задачи

	43.50	1.50	9.00	10.50
2	1.00	0.00	0.00	1.00
1	0.25	0.25	-1.50	-2.25
3	0.75	-0.25	2.50	2.25

Решение задачи:

$\lambda_1 = 0.25$; $\lambda_2 = 1.0$; $\lambda_3 = 0.75$.

Здесь $\lambda_1 = \lambda_{11}$; $\lambda_2 = \lambda_{21}$; $\lambda_3 = \lambda_{13}$.

(3) Координирующий сигнал $\pi_к = 1.5$.

Тогда $\pi^т_{к1} = 1.5 \times \| \, 1 \; 1 \, \| = \| \, 1.5 \; 1.5 \, \|$;

$\pi^т_{к2} = 1.5 \times \| \, 1 \; 1 \, \| = \| \, 1.5 \; 1.5 \, \|$.

Эти сигналы поступают в подсистемы.

(4) Номер итерации r = r + 1 = 4.

Производится переход на шаг 1.

<u>Итерация</u> 4.

(1) При координирующих сигналах $\pi^{т}_{к1}$ = ‖ 1.5 1.5 ‖; $\pi^{т}_{к2}$ = ‖ 1.5 1.5 ‖. Производится модификация целевых функций по правилу $(c^{т}_{i} - \pi^{т}_{ki})\, x_i$, тогда

$f_1(x)$ = ‖ 3 - 1.5, 2 - 1.5 ‖ x = ‖ 1.5, 0.5 ‖ x $\to$ max;

$f_2(x)$ = ‖ 2 - 1.5, 3 - 1.5 ‖ x = ‖ 0.5, 1.5 ‖ x $\to$ max;

Оптимальное решение при таких целевых функциях (см. рис.16.4)

x^{*}_{13} = (6,0), (4,6);

x^{*}_{23} = (0,7), (3,6).

Эти решения уже сообщались Центру, тогда решение задачи Центра будет тем же самым, и координирующий сигнал не изменится. Следовательно, получено оптимальное решение.

$\lambda^{*}_{1} = \lambda_{11} = 0.25$; $\lambda^{*}_{2} = \lambda_{21} = 1.0$; $\lambda^{*}_{3} = \lambda_{13} = 0.75$.

Тогда

$x^{*}_{1} = \lambda_{11}\, x^{*}_{11} + \lambda_{13}\, x^{*}_{13}$ = 0.25 (4, 6) + 0.75 (6, 0) = = (16.5, 1.5);

$x^{*}_{2} = \lambda_{21}\, x^{*}_{21}$ = 1.0 (3,6) = (3.6).

Оптимальное значение целевой функции $f^{*}(x^{*}) = 43.5$.

Таким образом, 1-му цеху целесообразно выпускать продукцию A_1 в количестве 55 единиц, продукцию A_2 в количестве 15 единиц; 2-му цеху целесообразно выпускать продукцию A_3 в количестве 30 единиц и продукцию A_4 в количестве 60 единиц. Суммарный доход предприятия при таком плане выпуска продукции составит 4350.

Библиографический список

1. Алексеев В.М., Тихомиров В.М., Фомин С.В. Оптимальное управление. - М.: Наука, 1979. - 429 с.

2. Алексеев К.Б., Бебенин Г.Т. Управление космическими летательными аппаратами. - М.: Машиностроение, 1974.- 340 с.

3. Алексеев О.Г. Комплексное применение методов дискретной оптимизации. - М.: Наука, 1987. - 248 с.

4. Алиев Р.А. Методы интеграции в системах управления производством. - М.: Энергоатомиздат, 1989. - 271 с.

5. Атанс М., Фалб П. Оптимальное управление. - М.: Машиностроение, 1968. - 763 с.

6. Багриновский К.А. Основы согласования плановых решений. - М.: Наука, 1977. - 303с.

7. Баринов К.Н., Насонов В.П. Краевые задачи динамики полёта космических аппаратов. - Л.: ЛВИКА им. А.Ф. Можайского, 1970. - 211 с.

8. Болтянский В.Г. Математические методы оптимального управления. - М.: Наука, 1966. - 408 с.

9. Брайсон А., Хо-Ю-Ши. Прикладная теория оптимального управления. -М.: Мир, 1972. - 544 с.

10. Булавский В.А, Звягина Р.А., Яковлева М.А. Численные методы линейного программирования. - М.: Наука, 1977. - 308 с.

11. Бурков В.Н., Кондратьев В.В. Механизмы функционирования организационных систем. - М.: Наука, 1981. - 383 с.

12. Вагнер Г. Основы исследования операций. Т.1.- М.: Мир, 1972. - 335 с.

13. Васильев Ф.П. Численные методы решения экстремальных задач. - М.: Наука, 1980. - 520 с.

14. Вентцель Е.С. Исследование операций. Задачи, принципы, методология - М.: Наука, 1988. - 208 с.

15. Военная системотехника и системный анализ. Учебник. /Под ред. Соколова Б.В. - СПб.: ВИКУ им. А.Ф. Можайского, 1999. - 496 с.

16. Воробьев Н.Н. Основы теории игр. - М.: Наука, 1984.- 495 с.

17. Воронов А.А. Введение в динамику сложных управляемых систем. - М.: Наука, 1985. - 352 с.

18. Габасов Р., Кириллова Ф.М. Качественная теория оптимальных процессов. - М.: Наука, 1971. - 508 с.

19. Гасс С. Линейное программирование. Методы и приложения. - М.: Наука, 1961. - 303 с.

20. Гилл Ф., Мюррей У. Численные методы условной оптимизации. М.: Мир, 1977. - 290 с.

21. Гермейер Ю.Б. Введение в теорию исследования операций. - М.: Наука, 1971. - 384 с.

22. Гермейер Ю.Б. Игры с непротивоположными интересами. - М.: Наука, 1976. - 327 с.

23. Гольштейн Е.Г., Юдин Д.Б. Новые направления в линейном программировании. - М.: Сов.радио, 1966. - 524 с.

24. Горелик В.А., Горелов М.А., Кононенко А.Ф. Анализ конфликтных ситуаций в системах управления. - М.: Радио и связь, 1991. - 288 с.

25. Давыдов Э.Г. Исследование операций. - М.: Высшая школа, 1990. - 383 с.

26. Данциг Дж. Линейное программирование, его применения и обобщения. - М.: Прогресс, 1965. - 600 с.

27. Дружинин В.В., Конторов Д.С. Введение в теорию конфликта. - М.: Радио и связь, 1989. - 288 с.

28. Дубов Ю.А., Травкин С.И., Якимец В.Н. Многокритериальные модели формирования и выбора вариантов систем. - М.: Наука, 1986. - 296 с.

29. Емеличев В.А., Ковалев М.М., Кравцов М.К. Многогранники, графы, оптимизация. - М.: Наука, 1981. - 344 с.

30. Ермольев Ю.М., Ляшко И.И., Михалевич В.С., Тюптя В.И. Математические методы исследования операций.- Киев.: Вища школа, 1979. - 312 с.

31. Ермольев Ю.М., Ястремский А.И. Стохастические модели и методы в экономическом планировании. - М.: Наука,1979.- 256с.

32. Ефремов Р.Н., Шестаков Н.Н., Юсупов Р.М. Прикладная математика. Вып.3. Методы динамической оптимизации и теории игр. - Л.: МО СССР, 1978. - 210 с.

33. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений. - М.: Мир,1976.- 165с.

34. Зайченко Ю.П. Исследование операций.- Киев: Вища школа, 1979. - 391 с.

35. Зангвилл У.И. Нелинейное программирование. - М.: Сов. радио, 1973. - 312 с.

36. Зойтендейк Г. Методы возможных направлений. – М.: ИЛ, 1963. – 176 с.

37. Иванов В.А., Фалдин Н.В. Теория оптимальных систем автоматического управления. - М.: Наука, 1981. - 336 с.

38. Иоффе А.Д., Исаев В.К. Особые управления и скользящие режимы. //Математика на службе инженера (Основы теории оптимального управления). - М.: Знание, 1973. - с.123-139.

39. Калинин В.Н. Теоретические основы управления подвижными объектами и операциями их обслуживания. - МО СССР, 1989. - 224 с.

40. Калинин В.Н., Резников Б.А. Теория систем и управления (структурно-математический подход). - Л.: ВИКИ им. А.Ф. Можайского, 1987. -417 с.

41. Калинин В.Н., Резников Б.А., Варакин Е.И. Теория систем и оптимального управления. Часть 1. - Л.: ВИКИ,1979.- 319 с.

42. Калинин В.Н., Резников Б.А., Варакин Е.И. Теория систем и оптимального управления. Часть 2. - МО СССР, 1987. - 589 с.

43. Калман Р., Фалб П., Арбиб М. Очерки по математической теории систем. - М.: Мир, 1971. - 400 с.

44. Канторович Л.В. Математические методы в организации и планировании производства. – Л.: ЛГУ, 1939. – 67 с.

45. Канторович Л.В., Гавурин М.К. Применение математических методов в вопросах анализа грузопотоков. – В кн. «Проблемы повышения эффективности работы транспорта» - М.: АН СССР, 1949. – с. 110-138.

46. Карлин С. Математические методы в теории игр, программировании и экономике. - М.: Мир, 1964. - 838 с.

47. Карманов В.Г. Математическое программирование. - М.: Наука, 1980. - 288 с.

48. Кини Р.Л., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения. - М.: Радио и связь,1981.- 560 с.

49. Ковалев М.М. Дискретная оптимизация (целочисленное программирование). - Минск: БГУ, 1977. - 191 с.

50. Корбут А.А., Финкельштейн Ю.Ю. Дискретное программирование. М.: Наука, 1969. - 368 с.

51. Кофман А. Введение в теорию нечетких множеств. - М.: Радио и связь, 1982. - 432 с.

52. Кротов В.Ф., Гурман В.И. Методы и задачи оптимального управления. - М.: Наука, 1973. - 448 с.

53. Крылов И.А., Черноусько Ф.Л. О методе последовательных приближений для решения задач оптимального управления. - М.: ЖВ и МФ. - 1962.- Т.2.- № 6.- С.1132-1139.
54. Кун Г. Венгерский метод решения задачи о назначениях. – В кн. «Методы и алгоритмы решения транспортной задачи». – М.:Госстатиздат, 1963.
55. Ларичев О.И. Объективные модели и субъективные решения. - М.: Наука, 1987. - 144 с.
56. Леллеп Я. Основы математической теории оптимального управления. - Тарту: ТГУ,1981.- 67 с.
57. Ли Э.Б., Маркус Л. Основы теории оптимального управления. - М.: Наука, 1972.- 576 с.
58. Линейные неравенства и смежные вопросы /сб.статей под ред. Г.У.Куна, А.У.Таккера/. - М.: Изд.иностр.лит-ры, 1959. - 469 с.
59. Лэсдон Л.С. Оптимизация больших систем. - М.: Наука,1975. - 432 с.
60. Ляшенко И.Н. и др. Линейное и нелинейное программирование. - Киев: Вища школа, 1975. - 371 с.
61. Математическое обеспечение управления подвижными объектами. /Резников Б.А. и др. - МО СССР, 1985. - 149 с.
62. Месарович М., Мако Д., Такахара И. Теория иерархических многоуровневых систем. - М.: Мир, 1973. - 344 с.
63. Моисеев Н.Н. Элементы теории оптимальных систем. - М.: Наука, 1975. - 528 с.
64. Нейман Дж., Моргенштерн О. Теория игр и экономическое поведение. - М.: Наука, 1970 - 708 с.
65. Нечеткие множества в моделях управления и искусственного интеллекта /под ред. Д.А. Поспелова. - М.: Наука, 1986. - 391 с.
66. Нечеткие множества и теория возможностей /под ред. Р.Р. Ягера. - М.: Радио и связь, 1986. - 408 с.
67. Обработка нечеткой информации в системах принятия решений. - М.: Радио и связь, 1989. - 304 с.
68. Орловский С.А. Проблемы принятия решений при нечеткой исходной информации. - М.: Наука, 1981. - 208 с.
69. Основы теории оптимального управления. / В.Ф.Кротов, Б.А.Лагоша, С.М.Лобанов и др. - М.: Высшая школа, 1990.- 430 с.
70. Оуэн Г. Теория игр. - М.: Мир, 1971. - 230 с.
71. Перегудов Ф.И., Тарасенко Ф.П. Введение в системный анализ. М.: Высшая школа, 1989.

72. Подиновский В.В., Гаврилов В.М. Оптимизация по последовательно применяемым критериям. - М.: Сов.радио, 1975.- 192 с.

73. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. - М.: Наука, 1982. - 256 с.

74. Понтрягин Л.С. Обыкновенные дифференциальные уравнения. - М.: Наука, 1970. - 332 с.

75. Понтрягин Л.С., Болтянский В.Г., Гамкрелидзе Р.В., Мищенко Е.Ф. Математическая теория оптимальных процессов. - М.: Наука, 1983.- 392 с.

76. Пропой А.И. Элементы теории оптимальных дискретных процессов. - М.: Наука, 1973.

77. Ройтенберг Я.Н. Автоматическое управление. - М.: Наука, 1978. - 551 с.

78. Романовский И.В. Алгоритмы решения экстремальных задач. - М.: Наука, 1977. - 352 с.

79. Саати Т. Целочисленные методы оптимизации и связанные с ними экстремальные проблемы. - М.: Мир, 1973. - 302 с.

80. Современное состояние теории исследования операций. / Под ред. Н.Н. Моисеева. - М. Наука 1979. - 464 с.

81. Соколов Б.В. Комплексное планирование операций и управление структурами в АСУ подвижными объектами. - МО СССР, 1992. - 232 с.

82. Сю.Д., Мейер А. Современное состояние теории управления. - М.: Машиностроение, 1972. - 544 с.

83. Финкельштейн Ю.Ю. Приближенные методы и прикладные задачи дискретного программирования. - М.: Наука. - 264 с.

84. Ху Т. Целочисленное программирование и потоки в сетях. – М.: Мир, 1974. – 520 с.

85. Цурков В.И. Декомпозиция в задачах большой размерности. - М.: Наука, 1981. - 352 с.

86. Юдин Д.Б. Задачи и методы стохастического программирования. - М.: Сов.радио, 1979. - 392 с.

87. Юдин Д.Б. Математические методы управления в условиях неполной информации. - М.: Сов.радио, 1974. - 400 с.

88. Юдин Д.Б., Гольштейн Е.Г. Линейное программирование (теория и конечные методы). - М.: Физматгиз, 1963. - 775 с.