

Министерство образования и науки Российской Федерации

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

ФАКУЛЬТЕТ ДИСТАНЦИОННОГО ОБУЧЕНИЯ (ФДО)

Л. П. Турунтаев

ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Учебное пособие

Томск
2016

УДК 519.8(075.8)
ББК 22.183я73
Т 888

Рецензенты:

Тарасенко В. Ф., докт. техн. наук, профессор кафедры
теоретической кибернетики Томского государственного университета;

Ехлаков Ю. П., докт. техн. наук, профессор, зав. кафедрой
автоматизации обработки информации ТУСУР.

Турунтаев Л. П.

Т 888 Исследование операций : учебное пособие / Л. П. Турунтаев. — Томск :
ФДО, ТУСУР, 2016. — 138 с.

Учебное пособие предназначено для студентов вузов, обучающихся с применением дистанционных образовательных технологий по направлениям и специальностям, связанным с применением математических методов и моделей в управлении организационными системами, созданием систем поддержки принятия решений.

УДК 519.8(075.8)
ББК 22.183я73

© Турунтаев Л. П., 2016
© Оформление.
ФДО, ТУСУР, 2016

ОГЛАВЛЕНИЕ

Введение	5
1 Исследование операций и теория принятия решений	7
1.1 Основные понятия	7
1.2 Методология исследования операций	10
1.2.1 Принципы исследования операций	10
1.2.2 Основные этапы операционного исследования	11
1.3 Классификация задач исследования операций	13
2 Задачи линейного программирования	17
2.1 Типовые модели задач линейного программирования	17
2.1.1 Задача использования ресурсов	18
2.1.2 Транспортная задача линейного программирования	20
2.1.3 Задача о назначениях	22
2.2 Общая постановка задачи линейного программирования, ее геометрическая интерпретация	23
2.2.1 Общая постановка задачи	23
2.2.2 Каноническая форма ЗЛП	24
2.2.3 Переход к каноническому виду	24
2.2.4 Геометрическая интерпретация ЗЛП	25
2.3 Симплексный метод	28
2.3.1 Идея симплекс-метода	28
2.3.2 Построение начального опорного плана	29
2.3.3 Прямой симплекс-метод	31
2.3.4 Двойственный симплекс-метод	34
2.3.5 Двухэтапный симплекс-метод	36
2.4 Двойственность задач линейного программирования	38
2.4.1 Прямая и двойственная задачи	38
2.4.2 Теоремы двойственности и их экономическое содержание	39
2.4.3 Интерпретация симплекс-таблиц	41
2.4.4 Анализ линейных моделей	44
2.5 Дробно-линейное программирование	48
3 Задачи линейного программирования транспортного типа	52
3.1 Транспортная задача линейного программирования	52
3.1.1 Математическая модель задачи	52
3.1.2 Определение начального опорного плана задачи	53
3.1.3 Распределительный метод	54
3.1.4 Метод потенциалов	57

3.2	Транспортная задача с промежуточными пунктами	60
3.3	Задача о назначениях	62
3.3.1	Математическая модель задачи	62
3.3.2	Венгерский метод решения задачи	63
3.3.3	Метод минимальных линий	66
3.4	Транспортные сети	67
3.4.1	Примеры сетевых транспортных задач	67
3.4.2	Минимизация сети	68
3.5	Задача о кратчайшем пути	70
3.6	Задача о минимальном потоке	75
4	Дискретные задачи линейного программирования	79
4.1	Классификация моделей и методов дискретного программирования. Примеры задач	79
4.2	Метод отсечения. Алгоритм Гомори	83
4.3	Метод ветвей и границ	86
4.3.1	Общая задача целочисленного программирования	86
4.3.2	Задача о коммивояжере	88
5	Нелинейное программирование	100
5.1	Особенности задач нелинейного программирования	100
5.2	Метод неопределенных множителей Лагранжа	102
5.3	Функция Лагранжа для задачи линейного программирования. Понятие седловой точки функции Лагранжа	105
5.4	Теорема Куна — Таккера	107
5.5	Метод линейной аппроксимации	108
6	Динамическое программирование	112
6.1	Особенности задач динамического программирования	112
6.2	Принципы динамического программирования. Функциональные уравнения Беллмана	113
6.3	Задача о рюкзаке	118
7	Задачи упорядочения	122
7.1	Особенности задач упорядочения	122
7.2	Сетевое планирование	122
7.2.1	Построение сетевого графика	122
7.2.2	Расчет параметров сетевого графика	125
7.2.3	Оптимизация распределения трудовых ресурсов	129
7.3	Составление расписаний	130
	Заключение	133
	Литература	134
	Глоссарий	136

ВВЕДЕНИЕ

Процессы принятия решений лежат в основе любой целенаправленной деятельности человека и занимают одно из центральных мест в управлении этой деятельностью. Несомненно, что опыт и интуиция руководителя здесь играют очень большую роль, но в силу особенностей таких систем и сложности решаемых ими проблем вероятность принятия неправильного управленческого решения достаточно высока. Поэтому и потери от этого становятся неприемлемо большими. В связи с этим возникла объективная потребность в подкреплении процесса принятия управленческих решений научно-обоснованными приемами и методами, в превращении его из искусства в науку, причем науку точную, использующую математические методы моделирования, исследования и поиска наилучших путей разрешения проблемных ситуаций.

В России первые шаги поиска оптимальных решений в системах организационного управления посредством составления математических моделей объектов управления были предприняты в середине 30-х годов прошлого столетия математиком-экономистом Л. В. Канторовичем. За целый ряд работ в этой области, «за вклад в теорию оптимального распределения ресурсов» Л. В. Канторович в 1975 году совместно с американскими учеными получил Нобелевскую премию. Датой зарождения научной дисциплины по количественному обоснованию принимаемых решений под названием «Исследование операций» считается середина 50-х годов XX столетия, зародилась она в Соединенных Штатах Америки.

Исследование операций — это наука, составная часть теории принятия решений, включающая совокупность научных методов количественного обоснования принимаемых решений, направленных на получение заданного результата (цели).

В данном пособии рассматриваются типовые задачи исследования операций, вопросы формализации деятельности объектов управления, модели, методы и алгоритмы поиска оптимальных решений.

Соглашения, принятые в книге

Для улучшения восприятия материала в данной книге используются пиктограммы и специальное выделение важной информации.



.....
Эта пиктограмма означает определение или новое понятие.
.....



.....
Эта пиктограмма означает теорему.
.....



.....
Пример
.....

Эта пиктограмма означает пример. В данном блоке автор может привести практический пример для пояснения и разбора основных моментов, отраженных в теоретическом материале.



.....
Контрольные вопросы по главе
.....

Глава 1

ИССЛЕДОВАНИЕ ОПЕРАЦИЙ И ТЕОРИЯ ПРИНЯТИЯ РЕШЕНИЙ

1.1 Основные понятия

Любая деятельность человека или коллектива людей преследует определенные цели. Саму деятельность коллектива по достижению какой-нибудь цели называют операцией [1–3].

Операция — это упорядоченная совокупность связанных взаимными отношениями действий (работ), направленных на достижение цели.

Цель — желаемый результат деятельности, достижимый в пределах некоторого интервала времени.

Задача — желаемый результат деятельности, достижимый за намеченный (заданный) интервал времени и характеризующийся набором количественных данных или параметров этого результата.

Таким образом, цель становится задачей, если указан срок ее достижения и конкретизированы количественные характеристики желаемого результата.

Проблема — это потенциальная цель (задача), для которой еще или не найдены альтернативные способы ее достижения, или не представляется возможным выделить ресурсы на наши альтернативы и проведение операции для ее решения, или то и другое вместе. Проблема всегда связана с несоответствием между действительным и желаемым результатами деятельности.

Проблемы могут возникнуть при следующих условиях:

- функционирование системы в данный момент либо в будущем не обеспечивает достижение поставленных целей;
- проблемы носят комплексный характер, и разрешение одних приводит к возникновению новых и, соответственно, изменению целей деятельности.

Проблема всегда связана с определенными условиями, которые обобщенно называют **ситуацией**. Условия (ситуация) — это ограничения, налагаемые социаль-

ной и природной средой и влияющие на исход решения. Совокупность проблемы и ситуации образует **проблемную ситуацию**. Одна и та же проблема, но в различных условиях, со своей проблемной ситуацией может быть решена различными способами. Выявление и описание проблемной ситуации дает исходную информацию для постановки задачи принятия решений.

Управление деятельностью по устранению проблем, носящих, как правило, сложный характер отношений, осуществляется системами организационного управления (в литературе встречаются названия: человеко-машинные системы, социально-экономические, социотехнические и другие).

Под системой организационного управления будем понимать систему организации труда работающих (цех, предприятие, объединение, отрасль и т. п.), состоящую из двух подсистем: субъекта управления (аппарат управления) и объекта управления (рис. 1.1).

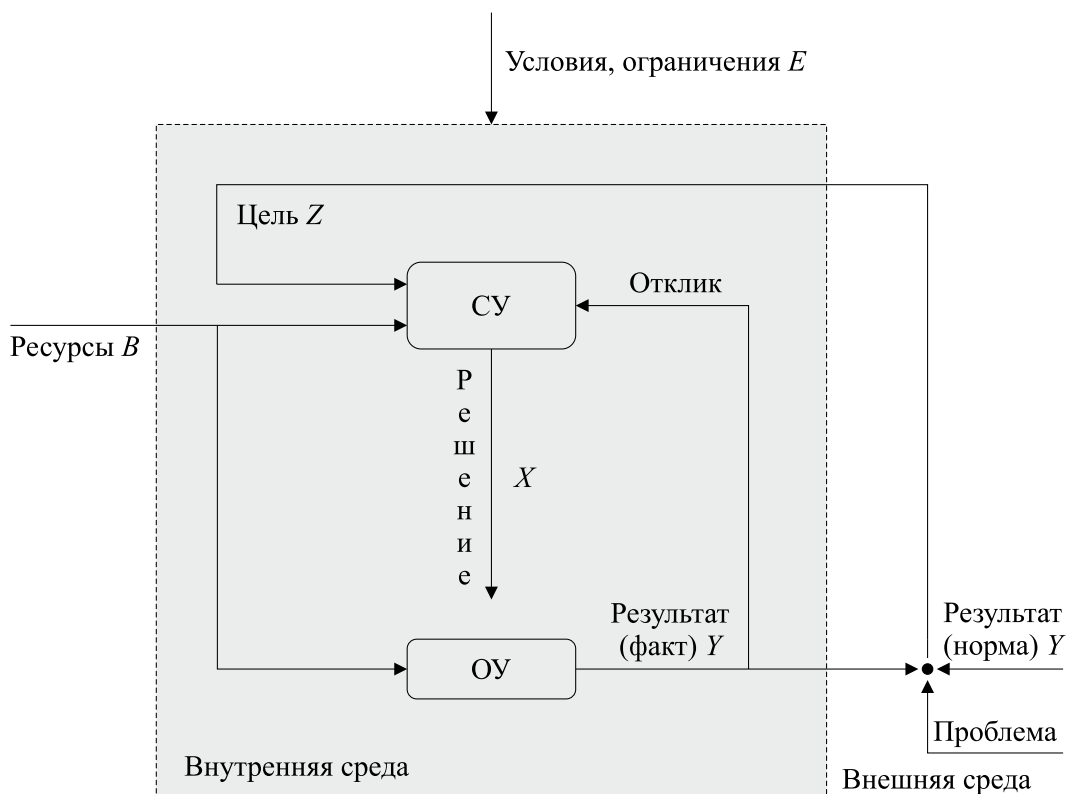


Рис. 1.1 – Система организационного управления

Из всего множества определений системы управления [4–6] отметим следующие:

- 1) система есть конечное множество функциональных элементов (подсистем) и отношений между ними, выделяемое из среды в соответствии с определенной целью, в рамках определенного временного интервала;
- 2) система есть средство выработки и реализации решений X по использованию ресурсов B , в условиях E , для достижения цели Z .

Конечным результатом деятельности субъекта управления является **решение**, которое представляет собой предписание к действию объекту управления. В качестве решения может быть предложен план работы, вариант проекта, распоряжение

и т. п. Процесс принятия решения связан с мыслительной деятельностью человека, протекающей во времени и направленной на поиск путей разрешения проблемной ситуации, а также проявлением волевого акта **выбора** конкретного пути решения, лучшего с точки зрения лица, принимающего решения (ЛПР). Выбор, или **принятие решения**, есть действие над множеством **альтернатив** (взаимоисключающих вариантов решений), результатом которого является подмножество отобранных альтернатив, представленное в виде одной альтернативы либо нескольких эффективных несравнимых альтернатив.

Выбор решений возможен, если имеется способ сравнения альтернатив между собой и определения их предпочтительности, т. е. имеется критерий предпочтения. **Предпочтение** — это интегральная оценка альтернатив качества решений, основанная на объективном анализе и субъективном понимании экспертов и ЛПР ценности соответствующих альтернатив.

Критерий — это правило, по которому отбираются альтернативы, он позволяет оценить альтернативы с точки зрения степени достижения цели. Если цель указывает желаемое состояние системы, отвечая на вопрос «Что нужно сделать?», то критерий позволяет определить эффективный способ достижения цели, отвечая на вопрос «Как нужно сделать?».

По характеру решаемых задач ИО близка к научной дисциплине «Теория принятия решений» (ТПР). В связи с этим целесообразно провести их сопоставление и выявить, в чем заключается различие между ними. Для задач ИО характерны следующие особенности [1–3, 8]:

- 1) объективный характер используемых моделей. Математические модели, используемые в исследовании операций, являются средством отражения объективно существующей реальности в объектах управления, как это имеет место в физике и других естественных науках;
- 2) заказы на проведение исследований дает руководитель, построение же модели объекта управления осуществляют аналитики, которые и ищут решение. Руководитель при этом может давать дополнительную информацию, но его роль здесь, в сущности, не отличается от роли других сотрудников организации. Главная задача руководителя — внедрить полученное решение;
- 3) существует объективный критерий успеха в применении методов исследования операций. Если проблема, требующая решения, ясна, критерий определен, то сразу видно, насколько найденное оптимальное решение лучше существующего.

Задачи, рассматриваемые ТПР, отличаются от задач исследования операций тем, что для их решения недостаточно объективных моделей и требуется привлечение дополнительной информации от лица, принимающего решение (ЛПР). Эта информация основана на опыте и интуиции ЛПР, она представляет точку зрения субъекта (группы субъектов) о предпочтительности решений, и поэтому она субъективна. В типичной ситуации с недостаточно определенными последствиями принимаемых решений, динамически меняющейся обстановкой только умение людей строить гипотезы и дополнять ими существующую информацию может спасти положение.

Следует заметить, что субъективность моделей, используемых в ТПР, вовсе не означает, что ЛПР может делать все, что ему захочется. Любые предпочтения ЛПР

должны находиться в рамках определенной рациональной системы, и субъективные решения, принимаемые ЛПР, сильно зависят не только от личности, но и от методов и процедур разработки и обоснования решений. Именно этими методами и процедурами и занимается ТПР.

1.2 Методология исследования операций

1.2.1 Принципы исследования операций

В качестве основных методологических принципов исследования операций ряд авторов выделяют следующие [1–3]:

- 1) системный подход как начальная фаза системного анализа определения проблемы и постановки задач;
- 2) комплексность и компетентность операционных групп исследования как необходимые условия всестороннего рассмотрения проблем выработки возможных решений и их оценки;
- 3) математическое моделирование как процесс построения моделей принятия решения на операцию.

Системный подход предполагает рассмотрение объектов исследования с позиции определения их как системы. Данные вопросы достаточно хорошо отражены в литературе по системному анализу [4, 5].

Системный подход как основной метод анализа и синтеза оптимальных решений реализуется посредством математического моделирования исследуемых объектов или операций.

Модель — это некоторое отражение оригинала. Математическая модель, представляющая из себя систему уравнений и ограничений, описывающих состояние исследуемой системы через ее входные и выходные параметры в любой момент времени, называется моделью без управления. Назовем ее Ω . Если ввести функцию предпочтения состояний системы Z (критерий оптимальности, целевая функция), то мы уже будем говорить об оптимизационной модели управления (Z, Ω) .

Модели без управления описывают процессы, течение которых в основных чертах определяется данной ситуацией (состоянием). Это, по существу, все модели анализа и прогнозирования систем: регрессионные, статические, балансовые, экономического роста и т. п.

Оптимизационные модели целесообразны и эффективны тогда, если они позволяют сгенерировать альтернативные решения и оценить их через критерии оптимальности с целью выбора наилучшей.

В решении сложных проблем используют имитационное моделирование. Оно используется там, где:

- 1) имеются весьма сложные объекты и процессы, для которых нет целостной теории, позволяющей применять классический математический аппарат;
- 2) математические модели хотя и могут быть построены, но не могут быть решены из-за огромного объема вычислений;

- 3) для получения решения необходимо учитывать эвристические или экспертные оценки.

В данном курсе исследования операций будут рассмотрены оптимизационные однокритериальные детерминированные задачи принятия решений.

1.2.2 Основные этапы операционного исследования

Выработка рекомендаций для ЛПР по выбору наилучшего решения является результатом операционного исследования, выполняемого операционной группой. В этом исследовании можно выделить следующие этапы [8]:

- 1) постановка задачи:
 - выявление проблемы;
 - формирование целей и критериев;
 - анализ проблемы и отбор факторов, описывающих ее;
 - построение математической модели;
- 2) поиск оптимального решения:
 - по отдельным критериям;
 - синтез оптимального (компромиссного) решения;
- 3) принятие и реализация решения:
 - принятие решения;
 - оценка полученного результата (проверка модели и оценка решения);
 - корректировка модели.

В сущности, этап постановки задачи (за исключением вопроса построения математических моделей) не относится к исследованию операций как теории математических моделей принятия решений, а является предметом системного анализа. Задача сначала формулируется с точки зрения заказчика (ЛПР), а затем происходит ее уточнение. При постановке задачи исследователь должен собрать следующую информацию:

- 1) кто принимает решения и каковы его (или их) цели;
- 2) на какие параметры задачи может влиять ЛПР (отбор управляемых переменных), каков диапазон этих параметров, каковы связи этих параметров между собой и с другими параметрами задачи;
- 3) на какие параметры задачи не может влиять ЛПР (отбор неуправляемых переменных), каковы их значения;
- 4) условия проведения операции.

Условия проведения операций связаны с уровнем информированности ЛПР о результатах реализации решений. С этой точки зрения задачи исследования операций делятся на детерминированные, вероятностные и неопределенные.

Задача называется детерминированной (условия определенности), если неуправляемые переменные операции точно известны и не меняются в ходе операции.

Задача является вероятностной (условия риска), если неуправляемые переменные операции — случайные величины, для которых известны их вероятностные характеристики.

Задача называется неопределенной (условия неопределенности), если нет точных сведений о значениях неуправляемых переменных и неизвестны законы их распределения.

В данном курсе рассматриваются задачи первой группы.

Математическую (формализованную) постановку задачи исследования операций в условиях определенности можно представить в виде оптимизационной модели. Математическая модель решения задачи служит для выяснения количественных оценок предполагаемых действий. Формализация задачи в принципе может иметь два крайних случая. В первом случае в распоряжении субъекта управления к моменту получения задачи имеется формальная модель, подходящая для описания возникшей ситуации; во втором — такой готовой модели нет, но есть время для ее составления. В практике же, как правило, наблюдается промежуточное положение, т. е. имеются модели, частично пригодные для формализации возникшей ситуации.

Как при составлении новой модели, так и при анализе имеющихся моделей субъекту управления необходимо решить следующие вопросы:

- уяснить характер задачи, ее структуру;
- выбрать математический аппарат, используемый для формализации;
- установить ограничения и допущения, принятые при составлении формальной модели;
- сравнить модель с реальной ситуацией.

В качестве основных количественных методов обоснования управленческих решений для ЗПР в условиях определенности являются методы математического программирования. Общая постановка однокритериальных детерминированных задач принятия решений полностью совпадает с общей постановкой задач математического программирования (ЗМП). Поэтому весь арсенал математических методов, разрабатываемых для решения ЗМП, может и должен быть использован для решения ЗПР этого класса.

Сделаем общую постановку задачи математического программирования (МП).

Пусть $x_1, x_2, \dots, x_n$ — переменные (неизвестные) величины, которые характеризуют систему управления; $Z(x_1, x_2, \dots, x_n)$ — целевая функция, цель управления системой; $g_i(x_1, x_2, \dots, x_n) R b_i, i = 1, 2, \dots, m$ — ограничения, накладываемые системой на неизвестные величины, где b_i — константы, R — возможные отношения $\geq, =, \leq$.

Необходимо найти вектор управления (план) $X = (x_1, x_2, \dots, x_n)$, который максимизирует (минимизирует) целевую функцию $Z(X)$ и удовлетворяет системе ограничений $g_i(X) R b_i, i = \overline{1, m}$. На неизвестные величины могут быть наложены условия неотрицательности $x_j \geq 0, j = \overline{1, n}$. Объединение всех ограничений, накладываемых на неизвестные величины, называют областью допустимых решений и обозначают буквой Ω , т. е. $X \in \Omega$. Таким образом, общая детерминированная модель математического программирования примет вид:

$$\max(\min): Z = Z(X), X \in \Omega.$$

Допустимый план $X \in \Omega$, доставляющий экстремальное значение целевой функции (критерию оптимальности), называется оптимальным.

В зависимости от вида функций $Z(X)$, $g_i(X)$ задачи МП делятся на несколько классов:

- 1) задачи линейного программирования, если $Z(X)$ и $g(X)$ линейны относительно переменных X ;
- 2) задачи нелинейного программирования, если хотя бы одна из этих функций нелинейна относительно X ;
- 3) задачи дискретного (целочисленного) программирования, если хотя бы одна из управляемых переменных, входящих в вектор X , должна принимать дискретные значения (например, целые);
- 4) задачи динамического программирования, если решение исходной задачи можно разложить на ряд более простых задач, для каждой из которых определены рекуррентные вычисления соответствующей управляемой переменной, входящей в вектор X .

Метод поиска оптимального решения существенно зависит от вида математической модели. В настоящее время наиболее разработаны методы решения задач линейного программирования, в рамки которого укладывается весьма широкий круг задач исследования операций.

Любая модель, в том числе и математическая, лишь частично отражает реальный процесс, поэтому необходима проверка системы ее соответствия моделируемому процессу. Проверка производится сравнением результатов, полученных на модели, с фактическими результатами действительно проведенной операции.

1.3 Классификация задач исследования операций

В условиях определенности задачи принятия решений, как правило, хорошо формализуются и описываются в терминах количественных переменных, и для их решения используются оптимизационные модели и аппарат математического программирования. Независимо от того, какой метод решения задачи используется, всегда отыскивается оптимальное или близкое к нему решение, максимизирующее критерий качества на модели (целевую функцию) при заданных условиях и ограничениях. Наиболее хорошо разработаны модели и алгоритмы решения на этих моделях для следующих классов задач:

- 1) распределения;
- 2) управления запасами;
- 3) массового обслуживания;
- 4) упорядочения и согласования;
- 5) выбора маршрута.

Распределительные задачи связаны с распределением ресурсов по работам, при котором минимизируются общие затраты (либо максимизируется общий доход). Они могут решаться методами линейного и динамического программирования. Яркими представителями распределительных задач являются задачи транспортные, о назначениях, использования ресурсов.

Задача управления запасами заключается в минимизации убытков, связанных с пополнением и хранением запасов и издержками из-за неудовлетворенного спроса. В результате решения получают ответ относительно размеров заказываемой партии, величины уровня запасов, точек размещения заказов и др.

Цель теории массового обслуживания — анализ процесса образования очередей «клиентами» при обслуживании, взаимосвязей между их основными характеристиками и выявление наилучших путей управления ими. В системах массового обслуживания присутствуют издержки, связанные с потерей клиентов из-за большой очереди или простоем оборудования. Задача сводится к минимизации всех видов издержек.

Содержанием задач упорядочения и координации является выбор дисциплины очереди. В качестве критерия оптимальности может быть время обслуживания, издержки по переналадке механизмов и др. Наиболее актуальными задачами являются задачи сетевого планирования и теории расписаний. В задачах сетевого планирования оптимизируются сроки выполнения всего комплекса операций (работ), представленного в виде сетевого графика, либо при заданных сроках минимизируются ресурсы на выполнение этих операций. В задачах теории расписаний формируется очередность операций, выполняемых одной машиной (задача директора), либо составляется расписание выполнения последовательности действий несколькими машинами. При решении задач сетевого планирования и теории расписаний широко применяется теория графов и комбинаторный анализ.

К задачам упорядочения тесно примыкают задачи выбора маршрута. На сети ищется маршрут доставки грузов нескольким потребителям либо в адрес одного, который минимизирует затраты на доставку. К данной группе задач выбора в качестве типичного представителя относят задачу коммивояжера.

Совокупность математических методов и алгоритмов со ссылкой на литературу, которые могут быть использованы для решения соответствующих классов задач исследования операций, представлена в таблице 1.1. В зависимости от постановки каждая из задач может решаться различными методами и соответственно один и тот же метод используется для решения ряда задач.



Контрольные вопросы по главе 1

1. В чем различие задач исследования операций и задач принятия решений?
2. Что является ключевым понятием в определении «система»?
3. Что такое проблемы и проблемная ситуация?
4. Почему любую проблему следует рассматривать с учетом влияния других проблем и явлений?
5. Каково соотношение целей и критериев для оценки альтернатив?
6. Дайте определение исследованию операций.
7. Назовите основные принципы исследования операций.
8. Укажите основные этапы операционного исследования.
9. В чем заключается сущность моделирования?
10. Дайте классификацию задач и методов исследования операций.

Глава 2

ЗАДАЧИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

2.1 Типовые модели задач линейного программирования

Методы и модели для решения задач, в которых необходимая информация однозначна и достоверна, называются детерминированными. На рисунке 2.1 приводится классификация детерминированных моделей задач исследования операций, рассматриваемых в данной главе и решаемых методами линейного программирования (дискретные ЗЛП рассматриваются в главе 4). Линейное программирование — это раздел науки «Математическое программирование», применяемый при разработке методов отыскания экстремума (максимума или минимума) линейной функции нескольких переменных при линейных ограничениях, наложенных на эти переменные.

Обратимся к классу задач (распределительные задачи) исследования операций, связанных с распределением ресурсов по работам, которые необходимо осуществить, с целью минимизации общих затрат на выполнение комплекса работ либо максимизации общего дохода, ожидаемого от выполнения этих работ.

Рассмотрим в качестве примера следующую ситуацию.

Две мебельные фабрики, входящие в одно объединение и расположенные в различных регионах, планируют выпуск продукции двух видов (столы и шкафы) каждой фабрикой. Для их изготовления необходимы ресурсы трех видов (пиломатериал, шурупы, краска), которые поставляются несколькими специализированными предприятиями, расположенными также в различных регионах. Доставка ресурсов на фабрики возможна различным транспортом с соответствующими затратами и в ограниченном количестве. Каждая из фабрик в силу своих особенностей характеризуется своими нормативными показателями расхода ресурсов на выпуск одной единицы продукции.

При заданных ценах на сырье и продукцию и при неограниченном сбыте и наличии сырья необходимо максимизировать прибыль работы объединения.

Сложную ситуацию можно упростить, если рассмотрим отдельно прослеживающиеся здесь задачи: использование ресурсов на фабрике для выпуска продукции (п. 2.1.1); подбор транспорта (задача о назначениях, п. 2.1.3) и доставка ресурсов до фабрик (транспортная задача, п. 2.1.2).

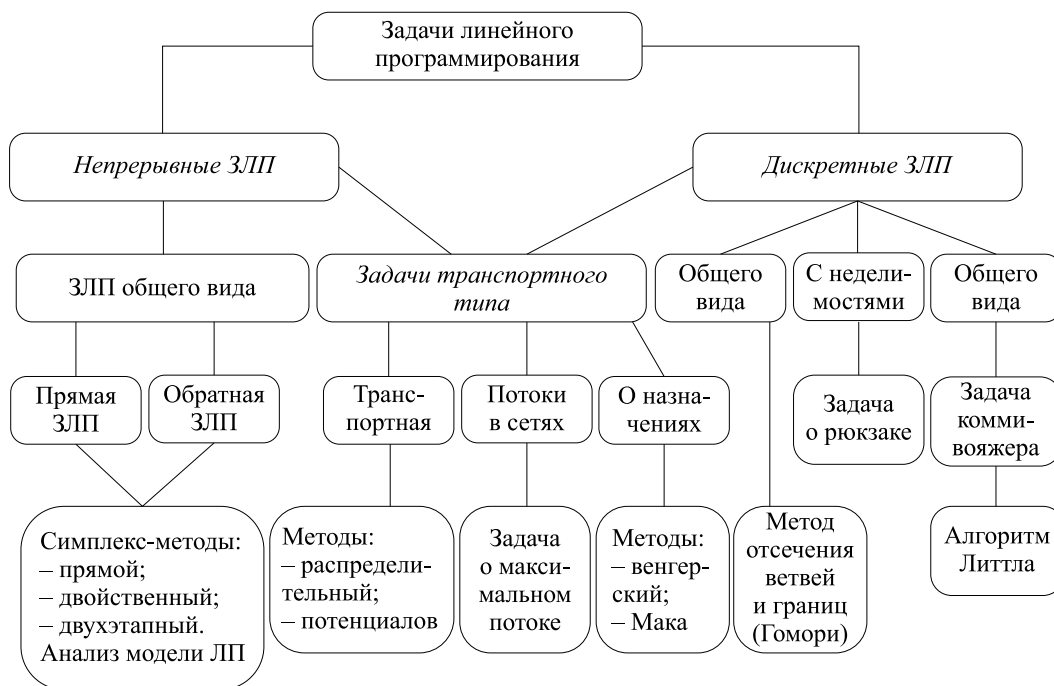


Рис. 2.1 – Классификация задач и методов линейного программирования

2.1.1 Задача использования ресурсов

Пусть мебельная фабрика изготавливает два вида продуктов: столы и шкафы. Для их производства используется три вида ресурсов (пиломатериал, шурупы, краска). Будем считать, что месячные запасы ресурсов ограничены: пиломатериал – величиной b_1 (м^3), шурупы – b_2 (кг), краска – b_3 (кг). Расходы соответствующих ресурсов на изготовление одной единицы соответствующих продуктов известны и задаются таблицей (матрицей) A . Доход от выпуска единицы соответствующей продукции задан: для стола он равен C_1 (руб./шт.), для шкафа – C_2 (руб./шт.). Требуется определить план выпуска продукции каждого вида, максимизирующий доход фабрики.

Построение математической модели

При построении математической модели для решения поставленной задачи необходимо дать ответы на следующие вопросы [2]:

- 1) для определения каких переменных (искомых величин) должна быть построена модель;
- 2) какие ограничения должны быть наложены на переменные, чтобы выполнялись условия, характерные для моделируемой системы;
- 3) в чем состоит цель, для достижения которой из всех допустимых значений переменных нужно выбрать те, которые будут соответствовать оптимальному (наилучшему) решению задачи.

Выразим словесно суть проблемы в разрезе поставленных вопросов и приступим к формализованному описанию задачи (рис. 2.2).

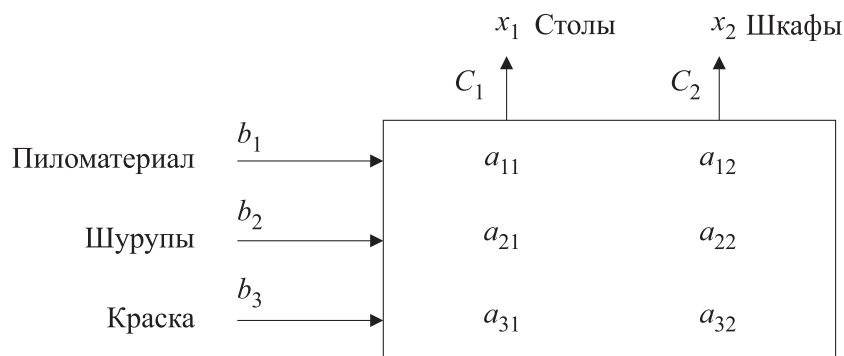


Рис. 2.2 – Формализованное описание задачи

Фабрике требуется определить объемы производства (в шт.) столов и шкафов, максимизирующие доход (в рублях) от их реализации, с учетом ограничений на расход исходных ресурсов.

Переменные. Так как нужно определить объемы производства каждого вида продукта, введем переменные:

- x_1 — месячный объем производства столов (шт.);
- x_2 — месячный объем производства шкафов (шт.).

Целевая функция. Если доход от реализации одного стола равен C_1 рублей, то от реализации столов в объеме x_1 штук месячный доход составит C_1x_1 рублей. Аналогично, месячный доход от реализации шкафов составит C_2x_2 рублей. Обозначив общий доход (в руб.) через Z , можно дать следующую математическую формулировку целевой функции: определить (допустимые) значения x_1 и x_2 , максимизирующие величину общего дохода

$$Z = C_1x_1 + C_2x_2 = \sum_{j=1}^2 C_jx_j.$$

Ограничения. При решении рассматриваемой задачи должны быть учтены ограничения на расход ресурсов. Пиломатериал идет на изготовление столов и шкафов. На один стол идет a_{11} м³ пиломатериала, тогда на столы в количестве x_1 штук потребуется $a_{11}x_1$ м³ пиломатериала. На изготовление шкафов в количестве x_2 штук потребуется $a_{12}x_2$ м³ пиломатериала. Всего пиломатериала потребуется $a_{11}x_1 + a_{12}x_2$ (м³). Расход его не должен превышать величины b_1 (м³).

Тогда ограничение на пиломатериал запишем в виде неравенства $a_{11}x_1 + a_{12}x_2 \leq b_1$ или $\sum_{j=1}^2 a_{1j}x_j \leq b_1$.

Аналогично запишем ограничения на другие ресурсы:

- на шурупы — $a_{21}x_1 + a_{22}x_2 \leq b_2$ или $\sum_{j=1}^2 a_{2j}x_j \leq b_2$;
- на краску — $a_{31}x_1 + a_{32}x_2 \leq b_3$ или $\sum_{j=1}^2 a_{3j}x_j \leq b_3$.

На переменные задачи x_1 и x_2 должны быть наложены условия неотрицательности и неделимости, т. е. введем ограничения $x_1 \geq 0$, $x_2 \geq 0$, x_1, x_2 — целые.

Итак, математическую модель задачи можно записать следующим образом. Определить месячные объемы производства столов x_1 и шкафов x_2 при которых достигается:

$$\max Z = C_1x_1 + C_2x_2 \text{ (целевая функция),}$$

при

$$\left. \begin{aligned} a_{11}x_1 + a_{12}x_2 &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 &\leq b_2 \\ a_{31}x_1 + a_{32}x_2 &\leq b_3 \\ x_1 &\geq 0, x_2 \geq 0 \\ x_1, x_2 &\text{ — целые} \end{aligned} \right\} \text{ (ограничения),}$$

или

$$\max Z = \sum_{j=1}^2 C_jx_j \quad \text{при} \quad \sum_{j=1}^2 a_{ij}x_j \leq b_i, \quad i = 1, 2, 3, \\ x_j \geq 0, \quad x_j \text{ — целые, } j = 1, 2.$$

Следует отметить, что с формальных позиций данная модель является линейной, потому что все входящие в нее функции (ограничения и целевая функция) линейны. Но линейный характер построенной модели должен предполагать наличие двух свойств — пропорциональности и аддитивности. Пропорциональность предполагает прямо пропорциональную зависимость между переменной и ее вкладом в целевую функцию и объем потребления ограниченных ресурсов. Например, прямая пропорциональность не будет иметь места, если ввести зависимость доходов фабрики от размера партии продаваемых продуктов. Аддитивность наблюдается в том, что доходы от переменных в целевой функции независимы, общий доход равен сумме доходов. Если фабрика производит два конкретных вида продукции, увеличение сбыта одного из которых отрицательно сказывается на объеме реализации другого, то такая модель не обладает свойством аддитивности.

2.1.2 Транспортная задача линейного программирования

Сделаем содержательную постановку транспортной задачи в следующем виде. На нескольких предприятиях деревообработки, расположенных в различных регионах, заготавливается пиломатериал, который необходимо по автомобильным дорогам доставить на ряд мебельных фабрик для производства мебели. Расстояния от предприятий заготовки до фабрик разные, соответственно различна и стоимость перевозки пиломатериала. Требуется так спланировать доставку материала на фабрику, чтобы общая стоимость перевозок была бы как можно меньше.

Конкретизируем задачу. Пусть имеется два предприятия (поставщики) и две фабрики (потребители). Известны возможности поставок пиломатериала поставщиками в месяц, соответственно $a_1 = 50 \text{ м}^3$ и $a_2 = 50 \text{ м}^3$, а потребности потребителей — соответственно $b_1 = 42 \text{ м}^3$ и $b_2 = 58 \text{ м}^3$. Стоимость перевозки 1 м^3 пиломатериала от первого поставщика первому потребителю $C_{11} = 2$ тыс. рублей, от первого поставщика второму потребителю $C_{12} = 1$ тыс. рублей, от второго — первому $C_{21} = 2$ тыс. рублей, от второго — второму $C_{22} = 3$ тыс. рублей. Определить план

перевозок продукции от поставщиков к потребителям, минимизирующий общую стоимость всех перевозок.

Представленная задача с равными возможностями поставщиков и потребностями потребителей ($a_1 + a_2 = b_1 + b_2$) называется сбалансированной. Построим математическую модель для этой задачи. Вначале представим модель задачи в виде транспортной сети (рис. 2.3).

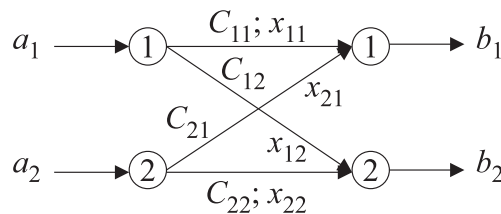


Рис. 2.3 – Сетевая модель транспортной задачи

Переменные. Пусть x_{11} — количество (м^3) перевозимого пиломатериала от пункта 1 в пункт 1; x_{12} — количество (м^3) перевозимого пиломатериала от пункта 1 в пункт 2; x_{21} , x_{22} — количество перевозимого пиломатериала из пункта 2 соответственно в пункт 1 и пункт 2.

Ограничения. Поставщики должны поставить:

- из пункта 1 — $x_{11} + x_{12} = a_1$;
- из пункта 2 — $x_{21} + x_{22} = a_2$ или $\sum_{j=1}^2 x_{ij} = a_i, i = 1, 2$.

Потребители должны получить:

- первый — $x_{11} + x_{21} = b_1$;
- второй — $x_{12} + x_{22} = b_2$ или $\sum_i x_{ij} = b_j, j = 1, 2$.

Кроме того, переменные $x_{ij} \geq 0, i = 1, 2; j = 1, 2$.

Целевая функция. В качестве целевой функции, подлежащей минимизации, выступают суммарные затраты на перевозку, определяемые формулой:

$$C_{11}x_{11} + C_{12}x_{12} + C_{21}x_{21} + C_{22}x_{22} = \sum_{i=1}^2 \sum_{j=1}^2 C_{ij}x_{ij}.$$

Окончательно модель задачи примет вид:

$$\min Z = \sum_{i=1}^2 \sum_{j=1}^2 C_{ij}x_{ij},$$

при ограничениях:

$$\begin{aligned} \sum_{j=1}^2 x_{ij} &= a_i, i = 1, 2, \\ \sum_{i=1}^2 x_{ij} &= b_j, j = 1, 2, \\ x_{ij} &\geq 0, \text{ для всех } i \text{ и } j. \end{aligned}$$

2.1.3 Задача о назначениях

Сделаем содержательную постановку задачи. В объединении находится n автомобилей, способных каждый перевозить в месяц Q_i тонн груза ($i = 1, 2, \dots, n$). С их помощью необходимо обеспечить перевозку грузов (шурупы) от поставщиков к потребителям по n маршрутам в количестве R_j тонн в месяц ($j = 1, 2, \dots, n$).

Распределить автомобили по маршрутам так, чтобы минимизировать суммарную величину неиспользуемой провозной способности. Конкретизируем задачу (рис. 2.4). Пусть имеется 4 автомобиля и 4 маршрута. Характеристики провозных способностей автомобилей соответственно равны $Q_1 = 30$ т, $Q_2 = 35$ т, $Q_3 = 5$ т, $Q_4 = 5$ т. Характеристики потребностей потребителей соответственно равны $R_1 = 25$ т, $R_2 = 32$ т, $R_3 = 5$ т, $R_4 = 4$ т.

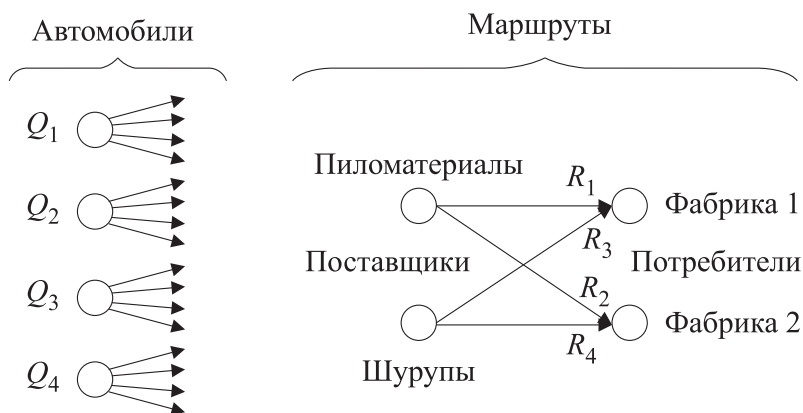


Рис. 2.4 – Схема маршрутов

Задача заключается в том, чтобы перевезти все грузы с минимальными издержками, для этого надо каждый автомобиль пустить по одному и только его маршруту. Понятно, если возможность автомобиля в перевозке груза ниже потребности потребителя этого груза, то на данный маршрут автомобиль не может быть назначен. Поэтому составим матрицу C , характеризующую издержки i -го автомобиля, в случае, если он будет назначен на j -й маршрут. Элементы маршрута будут равны:

$$C_{ij} = \begin{cases} Q_i - R_j, & \text{если } Q_i > R_j, \\ M, & \text{в противном случае (} M \text{ — большое число).} \end{cases}$$

Сделаем математическую постановку задачи о назначениях.

Переменные. В качестве переменной введем величину:

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-й автомобиль назначен на } j\text{-й маршрут;} \\ 0, & \text{в противном случае.} \end{cases}$$

Ограничения. Каждый автомобиль i должен быть назначен только один раз на любой из маршрутов:

$$x_{i1} + x_{i2} + x_{i3} + x_{i4} = 1 \text{ или } \sum_{j=1}^4 x_{ij} = 1, \quad i = \overline{1, 4}.$$

На каждый маршрут j должен быть назначен один из автомобилей:

$$x_{1j} + x_{2j} + x_{3j} + x_{4j} = 1 \text{ или } \sum_{i=1}^4 x_{ij} = 1, j = \overline{1, 4}.$$

Целевая функция. В качестве целевой функции, подлежащей минимизации, выступают суммарные издержки на перевозку.

Модель задачи о назначениях примет вид:

$$\min Z = \sum_{i=1}^4 \sum_{j=1}^4 C_{ij} x_{ij},$$

при ограничениях:

$$\sum_{j=1}^4 x_{ij} = 1, i = \overline{1, 4},$$

$$\sum_{i=1}^4 x_{ij} = 1, j = \overline{1, 4}, x_{ij} \in \{0, 1\}.$$

2.2 Общая постановка задачи линейного программирования, ее геометрическая интерпретация

2.2.1 Общая постановка задачи

Найти план $X = (x_1, x_2, \dots, x_n)$, который

$$\max(\min): Z(X) = \sum_{j=1}^n C_j x_j, \quad (2.1)$$

при системе ограничений:

$$\sum_{j=1}^n a_{ij} x_j = b_i, i = \overline{1, m_1}; \quad (2.2)$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, i = \overline{m_1 + 1, m_2}; \quad (2.3)$$

$$\sum_{j=1}^n a_{ij} x_j \geq b_i, i = \overline{m_2 + 1, m}; \quad (2.4)$$

$$x_j \geq 0, j = \overline{1, n}. \quad (2.5)$$

Линейная функция (2.1) называется целевой, множество планов X , удовлетворяющих системе ограничений (2.2)–(2.5), — множеством допустимых решений (обозначим буквой Ω), $X \in \Omega$. Допустимый план X , доставляющий целевой функции (2.1) экстремальное значение, именуется оптимальным.

2.2.2 Каноническая форма ЗЛП

Если все ограничения задачи (2.2)–(2.4) заданы в виде строгих равенств, то она называется ЗЛП в канонической или стандартной форме.

Примем за каноническую следующую форму записи ЗЛП:

$$\min Z(X) = \sum_{j=1}^n c_j x_j, \quad (2.6)$$

$$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i = \overline{1, m}, \quad (2.7)$$

$$x_j \geq 0, \quad j = \overline{1, n}. \quad (2.8)$$

В дальнейшем будем считать, что:

- 1) уравнения системы (2.7) линейно независимы, т. е. ни одно из них не может быть представлено как линейная комбинация остальных;
- 2) система (2.7)–(2.8) совместна, т. е. система уравнений непротиворечива и имеется совместная область допустимых решений Ω .

Все известные алгоритмы поиска оптимальных решений опираются на постановку ЗЛП в канонической форме. Поэтому необходим переход от общей постановки ЗЛП к канонической форме.

2.2.3 Переход к каноническому виду

Для перехода от задачи (2.1)–(2.5) к каноническому виду вводятся неотрицательные дополнительные переменные $x_{n+1} \geq 0, i = \overline{m_1 + 1, m}$, которые прибавляются к правым частям ограничений (2.3) и вычитаются из правых частей ограничений (2.4). В целевую функцию (2.1) дополнительные переменные вводятся с коэффициентами, равными нулю. Если в задаче, подлежащей решению, целевая функция $Z(X)$ максимизируется, то для приведения ее к виду (2.6) следует заменить в целевой $Z(X)$ знаки перед коэффициентами C_j на противоположные.

Очевидно, максимизация функции $Z(X)$ в области допустимых решений $X \in \Omega$ эквивалентна задаче минимизации функции — в той же области. При этом:

$$\max_{X \in \Omega} \sum_{j=1}^n c_j x_j = - \min_{X \in \Omega} \left(- \sum_{j=1}^n c_j x_j \right).$$

При переходе от исходной задачи (2.1)–(2.5) к ЗЛП в канонической форме множество допустимых решений Ω за счет введения дополнительных переменных расширяется до области $\Omega', \Omega \subset \Omega'$. Оптимальному решению исходной задачи соответствует оптимальное решение канонической (расширенной) задачи, т. к. множество допустимых решений Ω исходной задачи есть проекция множества допустимых решений Ω' расширенной задачи на подпространство исходных переменных [8].

2.2.4 Геометрическая интерпретация ЗЛП

Множество точек, удовлетворяющих условию (2.2), называется гиперплоскостью в n -мерном пространстве. Множество точек, удовлетворяющих условиям (2.3)–(2.5), называется полупространством в n -мерном пространстве.

В линейной алгебре показано, что гиперплоскости и полуплоскости образуют выпуклые множества, а пересечение (общая область) выпуклых множеств образует также выпуклое множество.

В случае двух переменных область допустимых решений Ω превращается в многоугольную выпуклую область, а в случае ее ограниченности — в выпуклый многогранник на плоскости (рис. 2.5).

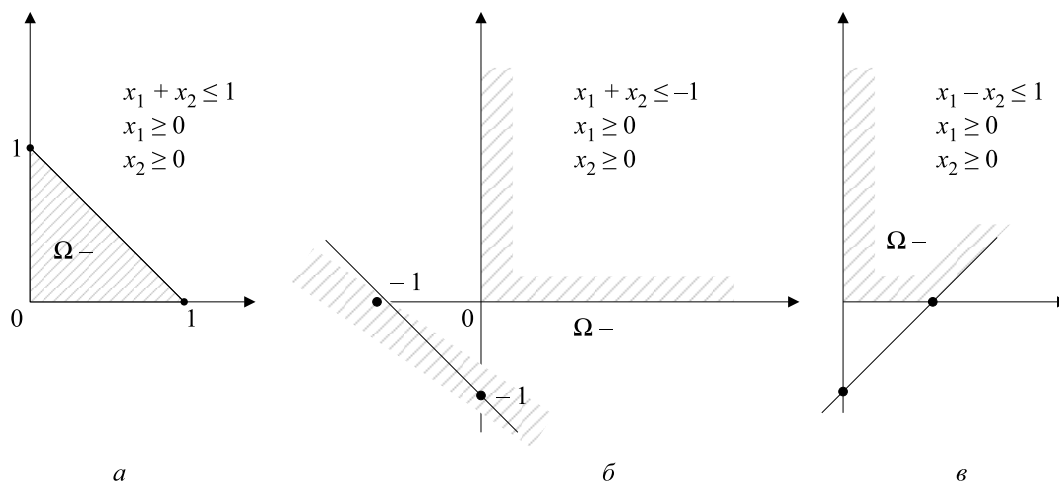


Рис. 2.5 – Многоугольные области: а) ограниченное множество; б) пустое множество; в) неограниченное множество

Перейдем к геометрической интерпретации целевой функции (2.1). Если зафиксировать значение Z , то целевая функция будет представлять некоторую гиперплоскость, во всех точках которой значение целевой функции не меняется. При изменении Z получим семейство гиперплоскостей уровня цели (при $n = 2$ получим семейство параллельных прямых). Вектор градиентного направления функции цели, составленный из коэффициентов при переменных x_j , а именно $\vec{C} = (c_1, c_2, \dots, c_n)$ — это вектор, указывающий направление наискорейшего возрастания целевой функции. Элементами этого вектора являются частные производные от функции Z по ее аргументам x_j : $\partial Z / \partial x_j = c_j$. Противоположное направление вектора $\vec{C}$ есть направление наискорейшего убывания целевой функции или антиградиентное. Семейство гиперплоскостей уровня цели располагается перпендикулярно вектору градиентного направления.

Графический способ решения ЗЛП в случае двух переменных

Конкретизируем задачу, рассмотренную в пп. 2.1.1, представив исходные данные в таблице 2.1 (данные приводятся условные).

Запишем модель задачи с приведенными данными:

$$\max Z = 500x_1 + 750x_2, \quad (2.9)$$

при ограничениях:

$$0,06x_1 + 0,07x_2 \leq 42, \quad (2.10)$$

$$0,04x_1 + 0,085x_2 \leq 34, \quad (2.11)$$

$$0,035x_1 + 0,12x_2 \leq 42, \quad (2.12)$$

$$x_1, x_2 \geq 0, \quad (2.13)$$

$$x_1, x_2 \text{ — целые.} \quad (2.14)$$

Таблица 2.1 – Исходные данные задачи

Ресурсы	Расход ресурсов на единицу продукции		Запас ресурсов
	Стол	Шкаф	
Пиломатериал, м ³	0,06	0,07	42
Шурупы, кг	0,04	0,085	34
Краска, кг	0,035	0,12	42
Цена единицы продукции, руб.	500	750	—

В дальнейшем ограничение (2.14) учитывать не будем, а решение задачи получим округлением найденных переменных задачи (2.9)–(2.14).

Алгоритм решения задачи графическим способом следующий.

1. Построим множество допустимых решений Ω задачи. Данное множество Ω образуется в результате пересечения полуплоскостей (ограничений) (2.10)–(2.13). На рисунке 2.6 множество допустимых решений показано в виде пятиугольника. Области, в которых выполняются соответствующие ограничения в виде неравенств, указываются стрелками, направленными в сторону допустимых значений переменных. Полученный многогранник Ω называют симплексом. Отсюда и название метода поиска оптимального решения табличным способом.

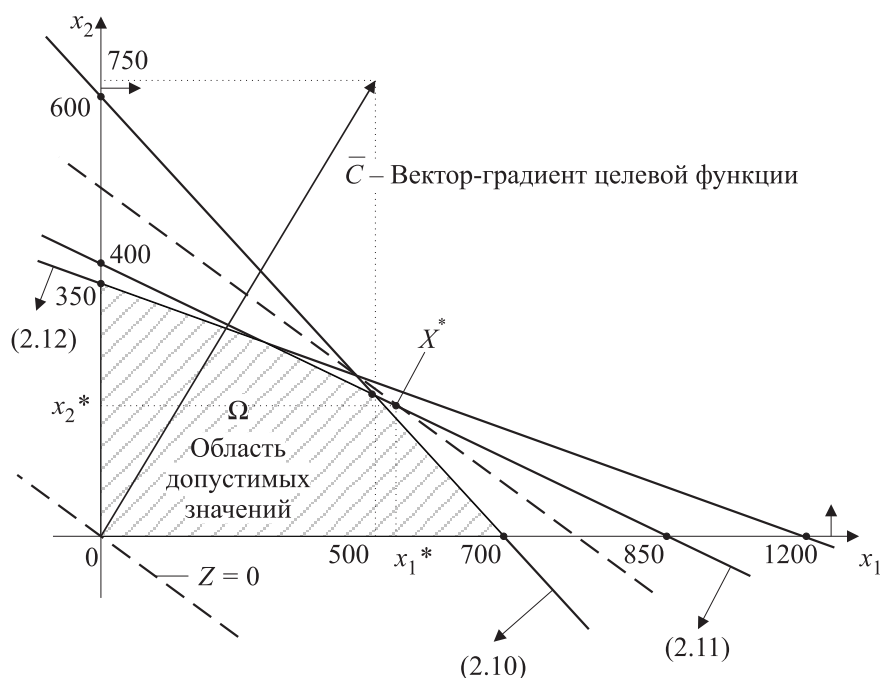


Рис. 2.6 – Геометрическая интерпретация ЗЛП

2. Построим вектор-градиент $\bar{C}$, составленный из производных целевой функции по переменным задачи, который указывает направление возрастания целевой функции по этим переменным. $\bar{C} = (C_1, C_2) = (500, 750)$. Начало этого вектора лежит в точке с координатами $(0, 0)$, а конец — в точке $(500, 750)$. Ряд параллельных штриховых линий, перпендикулярных вектору-градиенту, образуют множество целевых функций при произвольно выбранных значениях Z . При $Z = 0$ прямая (целевая функция) проходит через точку $(0, 0)$, а целевая функция Z принимает минимальное значение.

3. Перемещаем прямую, характеризующую доход Z в направлении вектор-градиента (для задачи $\max Z$), до тех пор, пока она не сместится в область недопустимых решений. На рисунке 2.6 видно, что оптимальному решению соответствует точка $X^* = (x_1^*, x_2^*)$. Так как точка X^* является точкой пересечения прямых (2.10) и (2.11), значения x_1^* и x_2^* определяются решением системы двух уравнений:

$$0,06x_1^* + 0,07x_2^* = 42;$$

$$0,04x_1^* + 0,085x_2^* = 34.$$

Решение указанной системы уравнений дает результат $x_1^* = 517,4$ и $x_2^* = 156,5$. Полученное решение означает, что месячный объем производства столов должен быть 517 шт., а шкафов — 156 шт. Доход, полученный в этом случае, составит $Z = 517 \cdot 500 + 156 \cdot 750 = 375\,500$ рублей.

Графический способ решения задачи ЗЛП со многими переменными

Задачу линейного программирования с числом переменных более двух можно решить графически, если в ее *канонической записи* число *неизвестных* n и число линейно независимых уравнений m связано соотношением $n - m \leq 2$. Запишем каноническую форму ЗЛП, рассмотренную выше. Для этого введем новые переменные x_3, x_4 и x_5 .

$$\max Z = 500x_1 + 750x_2 + 0 \cdot x_3 + 0 \cdot x_4 + 0 \cdot x_5, \quad (2.15)$$

$$0,06x_1 + 0,07x_2 + x_3 = 42, \quad (2.16)$$

$$0,04x_1 + 0,85x_2 + x_4 = 34, \quad (2.17)$$

$$0,035x_1 + 0,12x_2 + x_5 = 42, \quad (2.18)$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0. \quad (2.19)$$

Для данной ЗЛП число переменных $n = 5$, а число линейно независимых уравнений $m = 3$. Она и другие ЗЛП в канонической форме могут быть решены графически, если $n - m \leq 2$.

Выберем любые m неизвестных и выразим каждую из них через оставшиеся $(n - m)$ переменные. В нашем случае удобно взять переменные x_3, x_4 и x_5 и выразить их через x_1 и x_2 .

$$x_3 = 42 - 0,06x_1 - 0,07x_2 \geq 0,$$

$$x_4 = 34 - 0,04x_1 - 0,85x_2 \geq 0,$$

$$x_5 = 42 - 0,035x_1 + 0,12x_2 \geq 0.$$

Учитывая неотрицательность всех переменных, в том числе $x_3 \geq 0, x_4 \geq 0$ и $x_5 \geq 0$, а также зависимость последних от двух переменных x_1 и x_2 , можно графически показать решение расширенной задачи с проекцией на переменные x_1 и x_2 .

Полуплоскость $x_3 \geq 0$ (рис. 2.6) совпадает с ограничением (2.10), полуплоскость $x_4 \geq 0$ — с ограничением (2.11), а полуплоскость $x_5 \geq 0$ — с ограничением (2.12). Точка оптимума в координатах x_1 и x_2 образуется в результате пересечения полуплоскостей x_3 и x_4 : $x_1^* = 517,4$; $x_2^* = 156,5$. Соответственно значения переменных x_3 и x_4 будут нулевыми: $x_3^* = 0$; $x_4^* = 0$. Тогда из (2.18): $x_5^* = 42 - 0,035 \cdot 517,4 - 0,12 \cdot 156,5 = 5,1$.

Решением ЗЛП (2.15)–(2.19) будет вектор $X^* = (517,4; 156,5; 0; 0; 5,1)$, а $Z(X^*) = 375\,500$.

Геометрическое представление ЗЛП показывает:

- 1) множество допустимых решений Ω выпуклое;
- 2) оптимальное решение не существует, если множество Ω пустое или неограниченное в направлении перемещения семейства гиперплоскостей уровня цели поиска экстремума;
- 3) решение находится в одной из угловых точек (вершин) множества Ω допустимых решений, получивших название базисных;
- 4) для канонической ЗЛП базисные решения характеризуются вектором $X = (x_1, x_2, \dots, x_n)$, где n — число переменных задачи, в котором значения m переменных отличны от нуля, где m — число линейно-независимых уравнений задачи, число базисных переменных угловой точки множества Ω .

Для оптимального решения X^* рассмотренного примера базисными переменными стали переменные x_1 , x_2 и x_5 . Оставшиеся $(n - m)$ переменные называют небазисными или свободными. Их значения в угловой точке равны нулю.

Обратите внимание на то, что любая базисная переменная может быть выражена через небазисные и базисная переменная в модели (2.15)–(2.19) записывается один раз с коэффициентом единица.

2.3 Симплексный метод

2.3.1 Идея симплекс-метода

Рассмотренный графический способ решения задачи ЛП позволит понять идею методов оптимизации, в том числе и методов линейного программирования. Сущность всех методов МП заключается в том, чтобы вместо «слепого» перебора вариантов плана вести перебор выборочный, организованный, направленный на скорейшее, а в некоторых случаях и последовательное улучшение решения.

Экстремальное решение достигается не внутри области допустимых решений Ω , а на ее границе (рис. 3.2). Если быть еще точнее, то в одной из вершин угловых точек многоугольника, образованного в результате пересечения прямых, связанных с определенными ограничениями либо на отрезке между двумя соседними угловыми точками. Существует следующая основная теорема линейного программирования.



.....
 Если целевая функция ЗЛП достигает экстремального значения в некоторой точке области допустимых решений Ω , то она принимает это значение в угловой точке, если же более чем в одной угловой точке, то она принимает это же значение в любой точке из выпуклой линейной комбинации.

Так как экстремум обязательно достигается в одной или двух угловых точках допустимых планов, то нужно просто вычислить значения целевых функций во всех угловых точках (в нашем примере их пять) и выбрать ту из них, которой соответствует экстремальное значение. При большом числе переменных и при большом числе ограничений число угловых точек многогранника становится столь велико, что вычислить в каждой из них значение целевой функции, запомнить эти значения и сравнить между собой весьма проблематично даже для мощных ЭВМ. Поэтому нужно искать какой-то другой путь решения.

К точке оптимума можно подобраться последовательно, переходя от одной угловой точки к соседней, например от исходной (опорной) точки $X^0 = (x_1 = 0, x_2 = 0)$ к той соседней точке, которая ближе и быстрее приближает к X^* . Перебор точек решения по такой схеме предложил Р. Данциг и назвал его симплекс-методом [1–3]. Для нашего примера на первом шаге (итерации) от опорной точки X^0 мы перейдем по схеме симплекс-метода к точке X^1 с координатами $(700, 0)$ и на втором шаге перейдем к точке X^* (оптимальная симплекс-таблица).

Так как угловые точки характеризуются m базисными переменными, то переход от одной угловой точки к соседней возможно осуществить сменой в базисе только одной базисной переменной на переменную из не базиса.

Реализация симплекс-метода в силу различных особенностей и постановок задач ЛП имеет различные модификации.

2.3.2 Построение начального опорного плана

Пусть система ограничений такова:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i, \quad b_i \geq 0, \quad i = \overline{1, m}.$$

Добавив к левым частям дополнительные переменные $x_{n+i} \geq 0, i = \overline{1, m}$, получим каноническую (расширенную) задачу, стратегически эквивалентную исходной, с системой ограничений:

$$\sum_{j=1}^n a_{ij}x_j + x_{n+i} = b_i, \quad i = \overline{1, m}.$$

Тогда начальным опорным планом будет вектор:

$$X^0 = (\underbrace{0, 0, \dots, 0}_n, \underbrace{b_1, b_2, \dots, b_m}_m),$$

который удовлетворяет допустимости решения (он базисный, т. к. ненулевых элементов равно m , и опорный, т. к. все $(x_j \geq 0)$).

Пусть система ограничений такова:

$$\sum_{j=1}^n a_{ij}x_j \geq b_i, \quad b_i \geq 0, \quad i = \overline{1, m}.$$

Вычтя из левых частей дополнительные переменные $x_{n+i} \geq 0, i = \overline{1, m}$, получим расширенную задачу, стратегически эквивалентную исходной, с системой ограничений:

$$\sum_{j=1}^n a_{ij}x_j - x_{n+i} = b_i, \quad i = \overline{1, m}.$$

Однако теперь дополнительные переменные входят в левую часть ограничений с коэффициентами, равными минус единице. Поэтому план:

$$X^0 = (\underbrace{0, 0, \dots, 0}_n, \underbrace{-b_1, -b_2, \dots, -b_m}_m) -$$

не удовлетворяет условиям допустимости решения (он базисный, но не опорный).

Как в первом, так и во втором случае при добавлении дополнительных переменных (они же становятся базисными переменными) в систему ограничений эти же переменные вводятся в целевую функцию с коэффициентами, равными нулю: $C_{n+i} = 0, i = \overline{1, m}$. То есть в целевой функции при базисных переменных стоят нулевые коэффициенты, а при небазисных — коэффициенты $C_j, j = \overline{1, n}$. Пусть целевая функция устремлена на минимум. Тогда значение целевой функции может быть уменьшено, если в базис вводить ту переменную x_j , при которой коэффициент C_j в целевой функции имеет знак минус. И если все коэффициенты в целевой функции имеют знак плюс, то уменьшить ее значение не представляется возможным. Поэтому признаком оптимальности решения ЗЛП служат коэффициенты (оценки) в целевой функции при небазисных переменных.

В зависимости от выполнения условий оптимальности и допустимости применяются ту или иную схему решения ЗЛП [1–3, 8].

Методы решения ЗЛП разбиваются на две группы.

1. Методы последовательного улучшения решения. В основу их заложено движение от первоначальной точки (любое допустимое, но неоптимальное решение задачи (2.6)–(2.8) в канонической форме) к оптимальной точке за конечное число шагов (итераций). К этой группе относятся прямой симплекс-метод, метод потенциалов и другие.
2. Методы последовательного сокращения невязок. В основу их заложено движение от исходной условно-оптимальной точки, лежащей вне области допустимых решений (2.7), но удовлетворяющей признаку оптимальности решения, к оптимальной и допустимой точке. К этой группе относятся двойственный симплекс-метод, венгерский метод и другие. Все алгоритмы решения ЗЛП опираются на каноническую форму задачи. Поэтому число искомых переменных канонической задачи будет больше, чем исходной.

При выборе алгоритма решения задачи ЛП исходят из следующих данных:

- 1) находится минимум целевой функции;
- 2) сделана постановка ЗЛП в канонической форме;
- 3) определен начальный план задачи.

Тогда выбор алгоритма решения ЗЛП можно произвести согласно таблице 2.2.

Таблица 2.2 – Выбор алгоритма решения задачи

Проверка условия	Алгоритмы		
	Прямой симплекс-метод	Двойственный симплекс-метод	Двухэтапный симплекс-метод
На оптимальность решения ($c_j \geq 0$)	Не выполняются (есть $c_j < 0$)	Выполняются ($c_j \geq 0$)	Не выполняются (есть $c_j < 0$)
На допустимость решения ($x_j \geq 0$)	Выполняются ($x_j \geq 0$)	Не выполняются (есть $x_j < 0$)	Не выполняются (есть $x_j < 0$)

2.3.3 Прямой симплекс-метод

Рассмотрим прямой симплекс-метод решения задач ЛП табличным способом на следующем примере.



Пример 2.1

Минимизировать функцию $Z = -x_1 - x_2$, при ограничениях:

$$0,5x_1 + x_2 \leq 1;$$

$$2x_1 + x_2 \leq 2;$$

$$x_1, x_2 \geq 0.$$

Решение:

В канонической форме ЗЛП примет вид:

$$\min Z = -x_1 - x_2 + 0 \cdot x_3 + 0 \cdot x_4; \quad (2.20)$$

$$0,5x_1 + x_2 + x_3 = 1; \quad (2.21)$$

$$2x_1 + x_2 + x_4 = 2; \quad (2.22)$$

$$x_1, x_2, x_3, x_4 \geq 0. \quad (2.23)$$

Графическое представление задачи (2.20)–(2.23) показано на рисунке 2.7.

Начальной базисной опорной точкой задачи будет вектор $X^0 = (0; 0; 1; 2)$. Значение целевой функции в этой точке $Z(X^0) = 0$. Перенесем в целевой (2.20) переменную Z за знак равенства и данную задачу запишем в виде таблицы 2.3, называемой симплекс-таблицей (нулевая итерация).

В литературе описаны и другие формы записи симплекс-таблицы [1–3].

По симплекс-таблице всегда можно сказать, является ли найденное решение оптимальным. В данном случае решение $x_1 = 0; x_2 = 0; x_3 = 1; x_4 = 2$ не является наилучшим, так как можно ввести в базис одну из переменных x_1 или x_2 (при этих переменных стоят коэффициенты со знаком минус $c_1 = -1$ и $c_2 = -1$), уменьшив

значение целевой функции. Тогда вводя в базис одну из небазисных переменных x_1 или x_2 (увеличив ее значение), следует вывести из базиса переменные x_3 и x_4 (доведя ее значение до нуля). В прямом симплекс-методе рассматриваются последовательно вопросы:

- поиск переменной для включения в базис;
- поиск переменной для исключения из базиса;
- переход к новой канонической форме ЗЛП (к следующей итерации симплекс-таблицы).

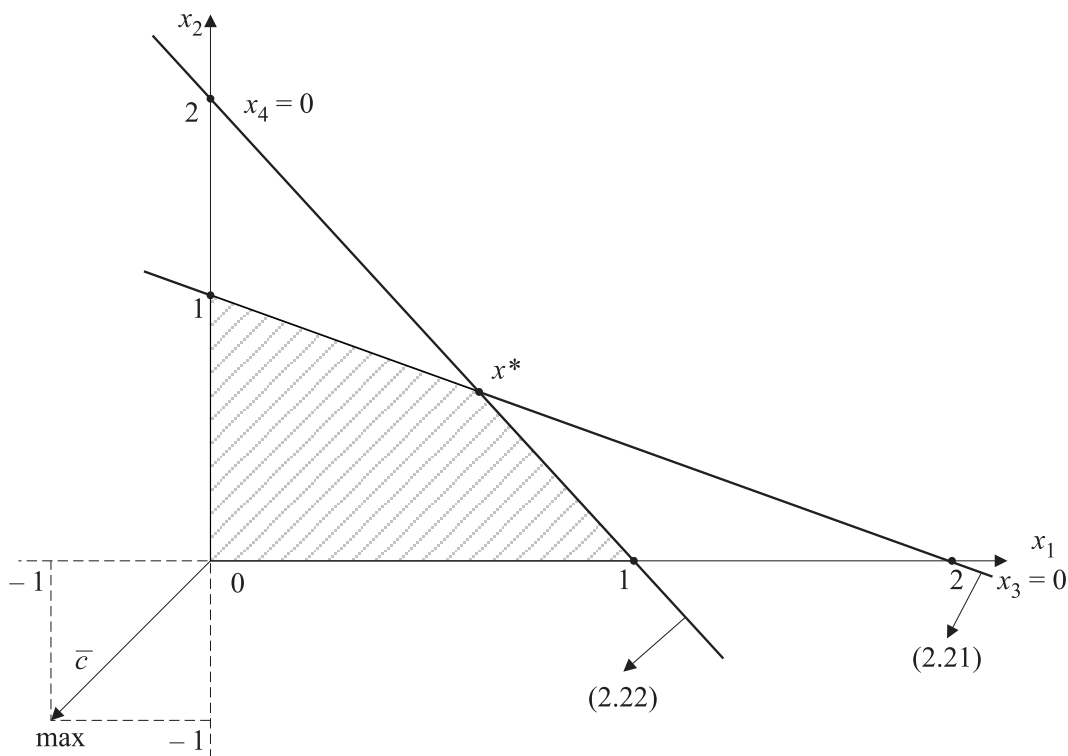


Рис. 2.7 – Графическое представление задачи

Таблица 2.3 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	Строка
0	$-z$	0	-1	0	0	0	0
	x_3	1	$\frac{1}{2}$	1	1	0	1
	x_4	2	2	1	0	1	2

Поиск переменной для включения в базис

Целесообразно включить в базис ту переменную, коэффициент при которой имеет наименьшее значение. Коэффициенты при небазисных переменных в неоптимальном решении имеют отрицательные значения. Пусть это будет переменная x_s , для которой $C_s = \min_j c_j$, $c_j < 0$, $j \notin \text{Б}$. В нашем примере $c_1 = c_2 = -1$, поэтому

включим в базис любую переменную x_1 или x_2 (пусть x_1). Столбец в симплекс-таблице с переменной x_s назовем ведущим столбцом, в нашем случае $s = 1$.

Поиск переменной для исключения из базиса

Если в базис включаем переменную x_1 , то это значит, что увеличиваем ее значение с нуля до каких-то определенных пределов. До каких? Обратимся к рисунку 2.7. Крайним значением для переменной x_1 будет единица, при этом переменная (прямая) x_4 в ограничении (2.22) примет значение, равное нулю. То есть из базиса выйдет x_4 , а ее место займет переменная x_1 . Из уравнения (2.21) определим значение $x_3 = 1 - 0,5 \cdot 1 = 0,5$. Таким образом, на следующей итерации (шаге) допустимым решением будет вектор $X^1 = (1; 0; 0,5; 0)$. Значение целевой функции в этой точке $Z(X^1) = -1$.

Не прибегая к графическому представлению задачи, определение предельного значения x_1 и определение переменной x_4 , которую следует вывести из базиса, можно провести на следующем распределении. Если вывести из базиса переменную x_3 , т. е. должно быть $x_3 = 0$, то из (2.21) следует $x_1 = b_1/a_{1s} = 1/0,5 = 2$. Если вывести из базиса переменную x_4 , т. е. сделать $x_4 = 0$, то из (2.22) $x_1 = b_2/a_{2s} = 1/1 = 1$. Получается, что значение $x_1 = 1$ или $x_1 = 2$. Но при $x_1 = 2$ в уравнении (2.22) переменная $x_4 = 1 - 2 - 0,5 \cdot 0 = -1$, что противоречит условию допустимости решения (2.23). Поэтому включаем в базис x_1 с наименьшим значением, которое определено из второго ограничения. В этом ограничении находится исключаемая переменная из базиса x_4 . В общем случае переменная x_s , включаемая в базис, может увеличиваться до значения:

$$\max x_s = \min_i \left(\frac{b_i}{a_{is}} \right), \quad i = 1, \dots, m, \quad a_{is} > 0.$$

Если этот максимум достигается в строке r , т. е. $x_s = b_r/a_{rs}$, то в этой строке базисная переменная обращается в нуль, т. е. выводится из базиса. Строку r называют ведущей строкой, а элемент a_{rs} — ведущим элементом. Если в ведущем столбце не найдутся положительные a_{is} , то это будет означать, что ЗЛП не имеет области допустимых решений.

Переход к новой канонической форме ЗЛП. В таблице 2.4 показаны переходы от нулевой итерации к последующим методам последовательного исключения вновь вводимой базисной переменной из других неведущих строк.

Новая строка на последующей итерации с вновь введенной базисной переменной получается путем деления элементов ведущей строки на ведущий элемент, относительно полученной строки далее производится исключение новой базисной переменной из других строк. В таблице 2.4 на итерации 1' указаны коэффициенты при базисных переменных, под которые осуществляется соответствующий переход. Ведущие элементы в таблице помечены звездочкой.

Расчет коэффициентов на очередной итерации можно производить по правилу четырехугольника [8].

Эта таблица на итерации 2 соответствует оптимальному решению $X^* = X^2 = (2/3; 2/3; 0; 0)$.

Значение целевой функции $Z(X^*) = -4/3$.

Таблица 2.4 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	Строка
0	$-z$	0	-1	-1	0	0	0
	x_3	1	$\frac{2}{3}$	1	1	0	1
	x_4	2	2^*	1	0	1	2
1'	$-z$		0		0		0
	x_3		0		1		1
	x_1		1		0		2
1''	$-z$	1	0	$-\frac{1}{2}$	0	$\frac{1}{2}$	0
	x_3	$\frac{1}{2}$	0	$\frac{3}{4}^*$	1	$-\frac{1}{4}$	1
	x_1	1	1	$\frac{1}{2}$	0	$\frac{1}{2}$	2
2	$-z$	$\frac{4}{3}$	0	0	$\frac{2}{3}$	$\frac{1}{3}$	0
	x_2	$\frac{2}{3}$	0	1	$\frac{4}{3}$	$-\frac{1}{3}$	1
	x_1	$\frac{2}{3}$	1	0	$-\frac{2}{3}$	$\frac{2}{3}$	2

2.3.4 Двойственный симплекс-метод

Рассмотрим двойственный симплекс-метод решения задачи ЛП на следующем примере.



Пример 2.2

Максимизировать функцию $Z = -x_1 - x_2$ при ограничениях:

$$0,5x_1 + x_2 \leq 1;$$

$$2x_1 + x_2 \geq 2;$$

$$x_1, x_2 \geq 0.$$

Решение:

В канонической форме ЗЛП примет вид:

$$\min Z' = -Z = x_1 + x_2 + 0 \cdot x_3 + 0 \cdot x_4; \quad (2.24)$$

$$0,5x_1 + x_2 + x_3 = 1; \quad (2.25)$$

$$2x_1 + x_2 - x_4 = 2; \quad (2.26)$$

$$x_1, x_2, x_3, x_4 \geq 0. \quad (2.27)$$

Графическое представление задачи показано на рисунке 2.8.

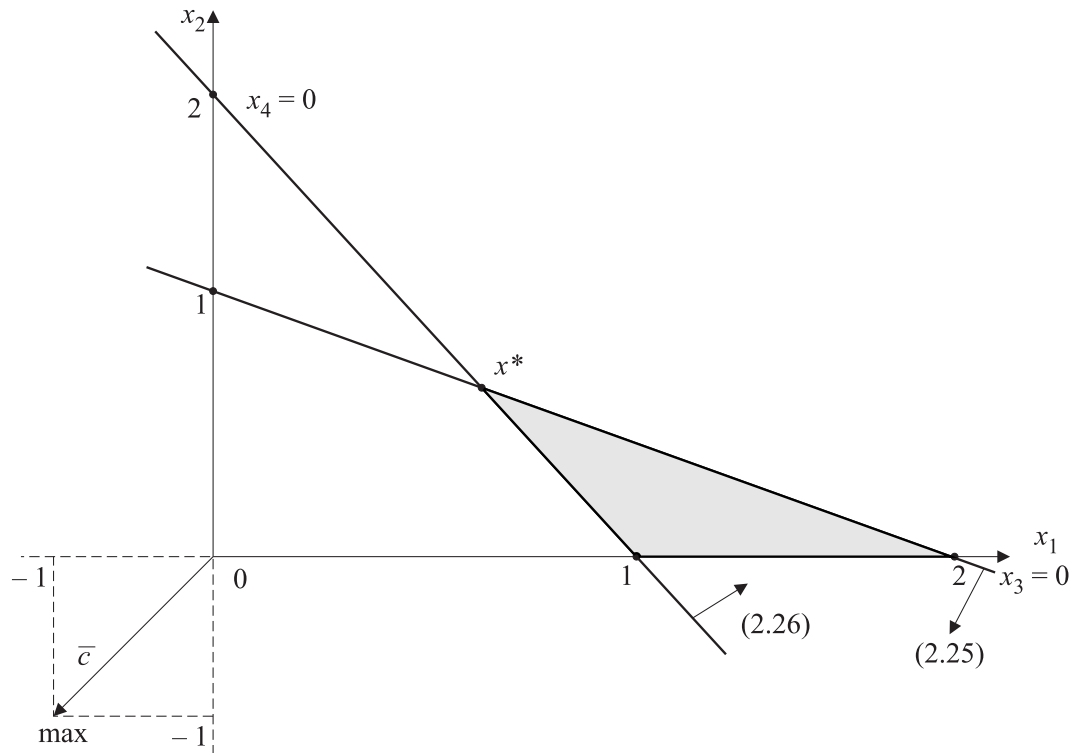


Рис. 2.8 – Графическое представление задачи

Составим симплекс-таблицу 2.5.

Таблица 2.5 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	Строка
0	$-z'$	0	1	1	0	0	0
	x_3	1	$\frac{1}{2}$	1	1	0	1
	x_4	-2	-2	-1	0	1	2

Нулевая строка в таблице 2.5 указывает на то, что признак оптимальности решения задачи выполнен (нет отрицательных коэффициентов). Однако начальное решение $X^0 = (0; 0; 1; -2)$ является отрицательным.

Попытаемся решить задачу (в противоположность симплекс-методу) последовательным движением от исходной недопустимой точки X^0 к X^* , рассматривая вопросы:

- поиск переменной для исключения из базиса;
- поиск переменной для включения в базис;
- переход к новой форме ЗЛП (последующей итерации решения задачи).

Поиск переменной для исключения из базиса

Из базиса исключается переменная, расположенная в строке r , имеющая наименьшее отрицательное значение. Если все переменные, расположенные в базисе, будут положительными, то вычисления заканчиваются, так как решение будет и оптимальным, и допустимым. В нашем примере исключаем переменную $x_4 = -2$.

Поиск переменной для включения в базис

Какую небазисную переменную включить в базис: x_1 или x_2 ? В принципе любую можно включить в базис с целью движения в область допустимых решений. Из графического представления задачи (рис. 2.8) видно, что при включении в базис переменной x_1 мы попадаем сразу в допустимую и оптимальную точку X^* . В литературе [1–3] показано, что к оптимальному решению можно добраться быстрее, если выбирать для включения в базис переменную x_s , такую, что для нее отношение $C_s/|a_{rs}|$ для всех элементов a_{rs} ведущей строки будет минимальным:

$$\frac{C_s}{|a_{rs}|} = \min_{j \in B} \frac{C_j}{|a_{rj}|}, \quad a_{rj} < 0. \quad (2.28)$$

Если все элементы $a_{rj} \geq 0$, то это будет означать, что задача не имеет допустимых решений.

В нашем примере минимальное отношение (2.28) достигается для переменной x_1 и равно $1/2$.

Решим задачу табличным способом (табл. 2.6).

Таблица 2.6 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	Строка
0	$-z'$	1	1	1	0	0	0
	x_3	1	$\frac{1}{2}$	1	1	0	1
	x_4	-2	-2^*	-1	0	1	2
1	$-z'$	-1	0	$\frac{1}{2}$	0	$\frac{1}{2}$	0
	x_3	$\frac{1}{2}$	0	$\frac{3}{4}$	1	$\frac{1}{4}$	1
	x_1	1	1	$\frac{1}{2}$	0	$-\frac{1}{2}$	2

Оптимальное решение: $X^* = (1; 0; 1/2; 0)$; $Z(X^*) = -z' = -1$.

.....

2.3.5 Двухэтапный симплекс-метод

Предположим, что при решении предыдущего примера (табл. 2.6) в базис включили бы не x_1 , а переменную x_2 , то получили бы на итерации 1 следующую таблицу 2.7. Нулевая строка в таблице 2.7 указывает на то, что признак

оптимальности решения задачи не выполнен, и промежуточное решение $X^1 = (0; 2; -1; 0)$ является недопустимым. Далее задачу можно решать двухэтапным симплекс-методом [8], методом больших штрафов [1] и другими [3].

Таблица 2.7 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	Строка
1	$-z'$	-2	-1	0	0	1	0
	x_3	-1	$-\frac{3}{2}$	0	1	1	1
	x_2	2	2	1	0	-1	2

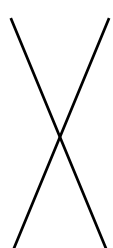
Рассмотрим двухэтапный симплекс-метод.

I этап

1. Вводим дополнительно по одной переменной, делая их базисными, в те уравнения, в которых не выполнялись условия допустимости. В нашем случае вводим переменную x_5 в строку (1), прежде изменив знаки на противоположные (табл. 2.8), и столбец под x_5 :

$$\frac{3}{2}x_1 - x_3 - x_4 + x_5 = 1.$$

Таблица 2.8 – Симплекс-таблица

Итерация	Базис	Значение	x_1	x_2	x_3	x_4	x_5	Строка
1	$-z'$	-2	-1	0	0	1	0	0
	x_5	1	$\frac{3}{2}^*$	0	-1	-1	1	1
	x_2	2	2	1	0	-1	0	2
	$-w$	-1	$-\frac{3}{2}$	0	1	1	0	3
2	$-z'$	$-\frac{4}{3}$	0	0	$-\frac{2}{3}$	$\frac{1}{3}$	$\frac{2}{3}$	0
	x_1	$\frac{2}{3}$	1	0	$-\frac{2}{3}$	$-\frac{2}{3}$	$\frac{2}{3}$	1
	x_2	$\frac{2}{3}$	0	1	$\frac{4}{3}^*$	$\frac{1}{3}$	$-\frac{4}{3}$	2
	$-w$	0	0	0	0	0	1	3
3	$-z'$	-1	0	$\frac{1}{2}$	0	$\frac{1}{2}$		0
	x_1	1	1	$\frac{1}{2}$	0	$-\frac{1}{2}$		1
	x_3	$\frac{1}{2}$	0	$\frac{3}{4}$	1	$\frac{1}{4}$		2

2. Вводим новую (фиктивную) целевую функцию W как сумму вновь вводимых дополнительных переменных, выраженную через небазисные переменные. В нашем случае:

$$W = x_5 = 1 - \frac{3}{2} \cdot x_1 + x_3 + x_4.$$

Вносим дополнительно строку 3 в таблицу 2.8 с фиктивной целевой функцией $-W - (3/2) \cdot x_1 + x_3 + x_4 = -1$.

3. Применяем прямой симплекс-метод для минимизации фиктивной целевой W с пересчетом всех коэффициентов. Первый этап заканчивается, если фиктивная целевая функция W обратится в нуль $W = 0$, а следовательно, и дополнительные переменные тоже будут с нулевыми значениями. Далее строка с фиктивной целевой функцией и столбцы с дополнительными переменными не рассматриваются. Если в результате минимизации целевой W получим оптимальное значение W , отличное от нуля $W \neq 0$, то это будет означать, что исходная ЗЛП не имеет допустимых решений.

II этап

4. Применяем прямой симплекс-метод для оптимизации основной целевой функции Z .

Оптимальное решение: $X^* = (1; 0; 1/2; 0)$; $Z(X^*) = -z' = -1$.

2.4 Двойственность задач линейного программирования

2.4.1 Прямая и двойственная задачи

Рассмотрим задачу линейного программирования (2.29), (2.31), (2.33) и проинтерпретируем ее как задачу использования ресурсов. Если для этой исходной ЗЛП (назовем ее прямой) ввести переменные y_i для оценки ресурсных ограничений (2.31) и сделать переход к математической постановке другой задачи (двойственной или обратной) вида (2.30), (2.32), (2.34), то решения прямой и двойственной задач находятся во взаимной зависимости, выраженной через соответствующие теоремы двойственности [1, 8].

Прямая задача			Двойственная задача	
$\max : Z(X) = \sum_{j=1}^n c_j x_j$	(2.29)	$\left \begin{array}{c} y_i \\ x_j \end{array} \right $	$\min : f(y) = \sum_{i=1}^m b_i y_i$	(2.30)
$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m$	(2.31)		$\sum_{j=1}^n a_{ij} y_i \geq c_j, \quad j = 1, \dots, n$	(2.32)
$x_j \geq 0, \quad j = 1, \dots, n$	(2.33)		$y_i \geq 0, \quad i = 1, \dots, m$	(2.34)

Переменные $y_1, y_2, \dots, y_m$ называют двойственными оценками. Пара задач (2.29), (2.31), (2.33) и (2.30), (2.32), (2.34) называется симметричной парой двойственных задач. Если среди ограничений прямой задачи имеются неравенства или на некоторые переменные не наложено условие неотрицательности, то, построив двойственную ей задачу, получаем пару несимметричных двойственных задач (2.35), (2.37), (2.39), (2.41) и (2.36), (2.38), (2.40), (2.42).

Прямая задача		Двойственная задача	
$\max : Z(X) = \sum_{j=1}^n c_j x_j \quad (2.35)$		$\min : f(y) = \sum_{i=1}^m b_i y_i \quad (2.36)$	
$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m_1 \quad (2.37)$	y_i	$\sum a_{ij} y_i \geq c_j, \quad j = 1, \dots, n_1 \quad (2.38)$	x_j
$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i = m_1 + 1, \dots, m \quad (2.39)$	y_i	$\sum a_{ij} y_i = c_j, \quad j = n_1 + 1, \dots, n \quad (2.40)$	x_j
$x_j \geq 0, \quad j = 1, \dots, n; \quad n_1 \leq n \quad (2.41)$		$y_i \geq 0, \quad i = 1, \dots, m_1; \quad m_1 \leq m \quad (2.42)$	

Сопоставляя модели прямой и двойственной задач, можно установить следующие взаимосвязи (табл. 2.9).

Таблица 2.9 – Взаимосвязь прямой и двойственной задач ЛП

Прямая ЗЛП	Двойственная ЗЛП
1. Целевая $Z(X) \rightarrow \max(\min)$	1. Целевая $f(y) \rightarrow \min(\max)$
2. Ограничение $\leq b_i (\geq b_i), i = 1, \dots, m_1$	2. Переменная $y_i \geq 0, i = 1, \dots, m$
3. Ограничение $= b_i, i = m_1 + 1, \dots, m$	2. Переменная y_i — любого знака, $i = m_1 + 1, \dots, m$
4. Переменная $x_j \geq 0, j = 1, \dots, n_1$	4. Ограничение $\geq c_j (\leq c_j), j = 1, \dots, n_1$
5. Переменная x_j — любого знака $j = n_1 + 1, \dots, n$	5. Ограничение $= c_j, j = n_1 + 1, \dots, n$

Очевидно, задача, двойственная двойственной, совпадает с исходной. Поэтому нет разницы, какую принять в качестве прямой, а какую — двойственной. Говорят о паре взаимно двойственных задач.

2.4.2 Теоремы двойственности и их экономическое содержание



Первая теорема двойственности

1. Малая теорема. Чтобы прямая задача $\max_x Z(X)$ и двойственная задача $\min_y f(y)$ имели оптимальные решения X^* и Y^* , необходимо и достаточно, чтобы существовали допустимые решения для каждой из них (рис. 2.9).

2. Основная (первая) теорема:

- если одна из пары двойственных задач имеет оптимальное решение, то и другая его имеет. Причем экстремальные значения их целевых функций равны $Z(X^*) = f(Y^*)$, в других случаях $Z(X) \leq f(Y)$;
- если целевая функция одной из задач не ограничена, то система условий другой противоречива (рис. 2.10).

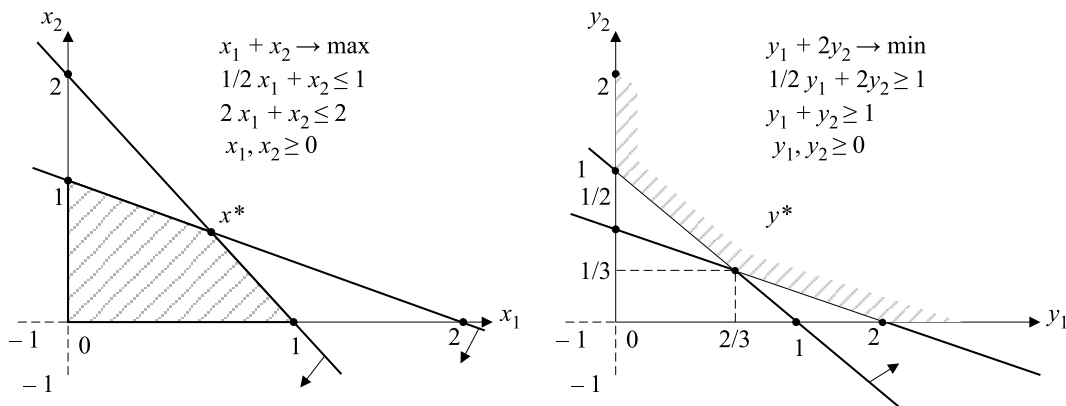


Рис. 2.9 – Пара двойственных задач (теорема 1)

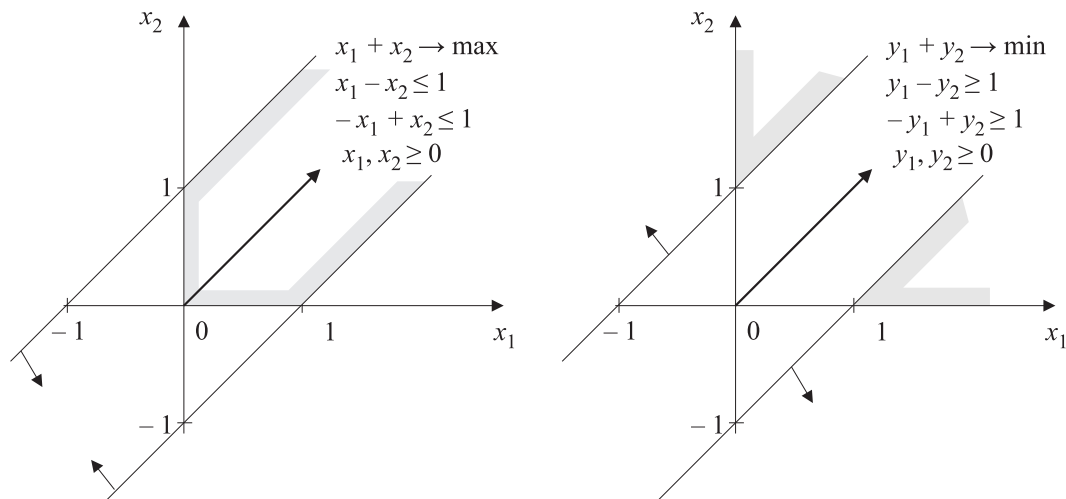


Рис. 2.10 – Пара двойственных задач

Установим экономический смысл первой теоремы двойственности (для задачи об оптимальном использовании ресурсов).

Если C_j — оценка (цена) реализации единицы выпускаемой продукции j ; x_j — объем выпускаемой продукции j ; b_i — объем имеющихся ресурсов i ; y_i — оценка (цена) ресурса i , то допустимый план производства X^* и вектор оценок ресурсов Y^* оказываются оптимальными тогда, когда доход от реализации всех видов продукции будет равен затратам на все виды ресурсов, используемых в производстве:

$$\sum_{j=1}^n c_j x_j = \sum_{i=1}^m b_i y_i.$$



Вторая теорема двойственности

Чтобы допустимые решения X^* и Y^* пары двойственных задач были оптимальными, необходимо и достаточно выполнить условия:

$$x_j^* \left(\sum_{i=1}^m a_{ij} y_i^* - c_j \right) = 0, \quad j = 1, \dots, n; \quad (2.43)$$

$$y_j^* \left(\sum_{i=1}^n a_{ij} x_j^* - b_i \right) = 0, \quad i = 1, \dots, m. \quad (2.44)$$

.....

Установим экономический смысл второй теоремы.

Из первого условия: если $x_j^* \neq 0$, то $\sum_{i=1}^m a_{ij} y_i^* = c_j$ — плата за ресурсы, предназначенные для изготовления продукции j , в оптимальном плане равна цене этой продукции. Иначе, если плата за ресурсы, идущие на единицу продукции, выше цены этой продукции $\left(\sum_{i=1}^m a_{ij} y_i^* = c_j \right) > 0$, то в оптимальном плане объем j -й продукции должен быть равен нулю $x_j^* = 0$.

Из второго условия: если $y_i^* = 0$, то $\sum_{j=1}^n a_{ij} x_j^* = b_i$ — ресурс i -го вида в производстве используется полностью, его остатка нет, он дефицитен. Оценка y_i характеризует степень дефицитности ресурса. Иначе, если i -й ресурс недефицитен, остается от производства продуктов $\left(b_i - \sum_{j=1}^n a_{ij} x_j^* \right) > 0$, то оценка дефицитности i -го ресурса $y_i^* = 0$.

Между переменными прямой и двойственной задач можно установить взаимнооднозначное соответствие. Если решением прямой задачи будет вектор $X = (x_1, \dots, x_n, x_{n+1}, \dots, x_{n+m})$, то переменные $x_1, \dots, x_n$ характеризуют объемы производства соответствующей продукции, а переменные $x_{n+1}, \dots, x_{n+m} \rightarrow x_{n+i}, i = \overline{1, m}$ — остатки ресурсов i -го вида. Решением двойственной задачи является вектор $Y = (y_1, \dots, y_m, y_{m+1}, \dots, y_{m+n})$. Переменные $y_1, \dots, y_m$ характеризуют степень дефицитности соответствующих ресурсов в объеме b_i единиц, а переменные $y_{m+1}, \dots, y_{m+n} \rightarrow y_{m+j}, j = \overline{1, n}$ — штраф за выпуск продукции j -го вида.

Для оптимальных решений пары двойственных задач условия второй теоремы (2.43)–(2.44) можно записать в виде:

$$\begin{aligned} x_j^* \cdot y_{m+j} &= 0, \quad j = \overline{1, n}; \\ y_i^* \cdot x_{n+i} &= 0, \quad i = \overline{1, m}, \end{aligned}$$

где

$$\begin{aligned} y_{m+j} &= \sum_{i=1}^m a_{ij} y_i^* - c_j, \quad j = \overline{1, n}, \\ x_{n+i} &= \sum_{j=1}^n a_{ij} x_j^* - b_i, \quad i = \overline{1, m}. \end{aligned}$$

2.4.3 Интерпретация симплекс-таблиц

Обратимся к задаче (2.20)–(2.23). В результате ее решения получили $X^* = (2/3; 2/3; 0; 0)$. Если бы перешли от прямой ЗЛП к двойственной и решили бы ее, то получили бы решение $Y^* = (2/3; 1/3; 0; 0)$. Величина $x_3 = 0$ характеризует

остаток ресурса 1, поэтому дефицитность его определяется через оценку $y_1 = 2/3 \rightarrow (y_1 \cdot x_3 = 0)$. Величина $x_4 = 0$ связана с оценкой $y_2 = 1/3 \rightarrow (y_2 \cdot x_4 = 0)$.

Величины $x_1 = 2/3$ и $x_2 = 2/3$ характеризуют объем выпуска продукции первого и второго видов, штраф за их выпуск оценивается через $y_3 = 0$ и $y_4 = 0$. Обратимся к оптимальной симплекс-таблице задачи (табл. 2.4) и покажем расположение двойственных оценок в ней (табл. 2.10).

Таблица 2.10 – Расположение оценок ЗЛП

Переменные прямой задачи		← Основные →		← Дополнительные →	
Базис	Значение	x_1	x_2	x_3	x_4
$-z$	$\frac{4}{3}$	$\textcircled{0}$	$\textcircled{0}$	$\textcircled{\frac{2}{3}}$	$\textcircled{\frac{1}{3}}$
$x_2 \leftarrow$	$\textcircled{\frac{2}{3}}$	0	1	$\frac{4}{3}$	$-\frac{1}{3}$
$x_1 \leftarrow$	$\textcircled{\frac{2}{3}}$	1	0	$-\frac{2}{3}$	$\frac{2}{3}$
Переменные обратной задачи		y_3	y_4	y_1	y_2
		← Дополнительные →		← Основные →	

Двойственные оценки задачи располагаются в строке z . Запишем её:

$$-z + 0 \cdot x_1 + 0 \cdot x_2 + 2/3 \cdot x_3 + 1/3 \cdot x_4 = 4/3 \text{ или}$$

$$z = 4/3 + (0 \cdot x_1 + 0 \cdot x_2 + 2/3 \cdot x_3 + 1/3 \cdot x_4) = -4/3 + (y_3 x_1 + y_4 x_2 + y_1 x_3 + y_2 x_4).$$

Оценки ресурсов (основные двойственные оценки) стоят перед дополнительными переменными прямой задачи. Оценки плана выпуска продукции (дополнительные двойственные оценки) стоят перед основными переменными прямой задачи. Основные двойственные оценки y_i , $i = \overline{1, 2}$ показывают величину изменения целевой функции при изменении запаса ресурса на 1 единицу:

$$\frac{dz}{dx_{2+i}} \approx \frac{\Delta z}{\Delta x_{2+i}} = \frac{\Delta z}{\Delta b_i} = y_i, \quad i = \overline{1, 2}.$$

Если одна из базисных переменных обращается в нуль (в решении число переменных, не равных нулю, станет меньше, чем число линейно независимых ограничений), то в этом случае говорят, что решение является вырожденным. С практической точки зрения специфика ситуации целиком объясняется наличием в модели по крайней мере одного избыточного ограничения. Покажем на примере [1].



Пример 2.3

$$\max : z = 3x_1 + 9x_2;$$

$$x_1 + 4x_2 \leq 8;$$

$$x_1 + 2x_2 \leq 4;$$

$$x_1, x_2 \geq 0.$$

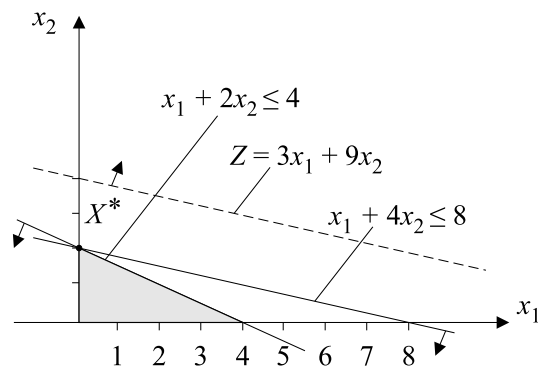


Рис. 2.11 – Графическое решение задачи

Решение:

Решим графически (рис. 2.11). На рисунке видно: первое ограничение $x_1 + 4x_2 \leq 8$ не влияет на решение, оно избыточное.

Оптимальная симплекс-таблица будет иметь следующий вид (табл. 2.11).

Таблица 2.11 – Оптимальное решение

Базис	Значение	x_1	x_2	x_3	x_4
$-z$	-18	0	0	$\frac{3}{2}$	$\frac{3}{2}$
x_2	2	0	1	$\frac{1}{2}$	$-\frac{1}{2}$
x_1	0	1	0	-1	2

Представляет интерес случай бесконечного множества решений на рисунке 2.12, который иллюстрирует условия задачи ЛП, особенность которой состоит в том, что прямая целевой функции параллельна прямой одного из ограничений. Это обуславливает наличие множества оптимальных решений на отрезке AB . В отличие от симплекс-таблиц с решением чисто в угловой точке (в этих таблицах число нулей в строке Z равно числу базисных переменных) в альтернативных оптимальных симплекс-таблицах число нулей в строке Z больше, чем число базисных переменных (табл. 2.12).

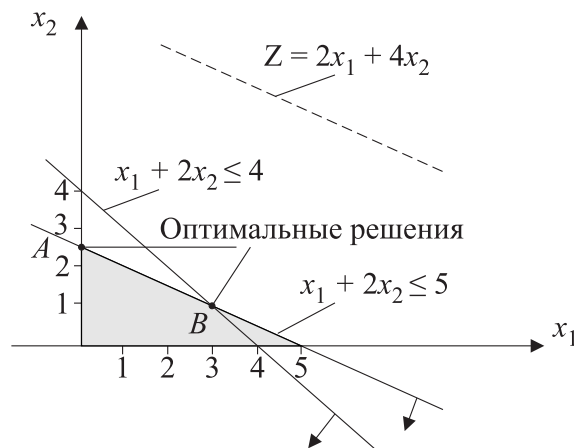


Рис. 2.12 – Графическое решение задачи

Таблица 2.12 – Альтернативные решения

	Базис	Значение	x_1	x_2	x_3	x_4
Оптимум в точке A (введем x_1)	$-z$	-10	0	0	2	0
	x_2	$\frac{5}{2}$	$\frac{1}{2}$	1	$\frac{1}{2}$	0
	x_4	$\frac{3}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$	1
Оптимум в точке B	$-z$	-10	0	0	2	0
	x_2	1	0	1	1	-1
	x_1	3	1	0	-1	2

Решение на отрезке AB можно определить как взвешенное среднее двух полученных оптимальных решений, соответствующих точкам A и B :

$$A: x_1 = 0; x_2 = 5/2;$$

$$B: x_1 = 3; x_2 = 1,$$

и полагая $0 \leq \alpha \leq 1$, получим:

$$\hat{x}_1 = \alpha \cdot (0) + (1 - \alpha) \cdot (3) = 3 - 3\alpha;$$

$$\hat{x}_2 = \alpha \cdot (5/2) + (1 - \alpha) \cdot (1) = 1 - 3\alpha/2.$$

2.4.4 Анализ линейных моделей

Данные в оптимальной симплекс-таблице позволяют делать всевозможный анализ линейной модели [1], в частности анализ чувствительности оптимального решения к изменению запасов ресурсов и вариациям коэффициентов целевой функции. Для рассмотрения примеров данных анализов из заключительной симплекс-таблицы обратимся к следующей модели [1], приведенной к стандартной форме:

$$\max : Z = 3x_1 + 2x_2 \text{ (прибыль)},$$

при ограничениях:

$$\begin{array}{rclclclcl} x_1 & + & 2x_2 & + & x_3 & & & = & 6; & (\text{ресурс 1}) \\ 2x_1 & + & x_2 & & & + & x_4 & = & 8; & (\text{ресурс 2}) \\ -x_1 & + & x_2 & & & & + & x_5 & = & 1; & (\text{ресурс 3}) \\ & & x_2 & & & & & + & x_6 & = & 2; & (\text{ресурс 4}) \end{array}$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0.$$

Оптимальная симплекс-таблица имеет вид, приведенный в таблице 2.13.

Таблица 2.13 – Оптимальная симплекс-таблица

Базис	x_1	x_2	x_3	x_4	x_5	x_6	Решение
z	0	0	$\frac{1}{3}$	$\frac{4}{3}$	0	0	$12\frac{2}{3}$
x_2	0	1	$\frac{2}{3}$	$-\frac{1}{3}$	0	0	$1\frac{1}{3}$
x_1	1	0	$-\frac{1}{3}$	$\frac{2}{3}$	0	0	$3\frac{1}{3}$
x_5	0	0	-1	1	1	0	3
x_6	0	0	$-\frac{2}{3}$	$\frac{1}{3}$	0	1	$\frac{2}{3}$

Графическое решение задачи показано на рисунке 2.13.

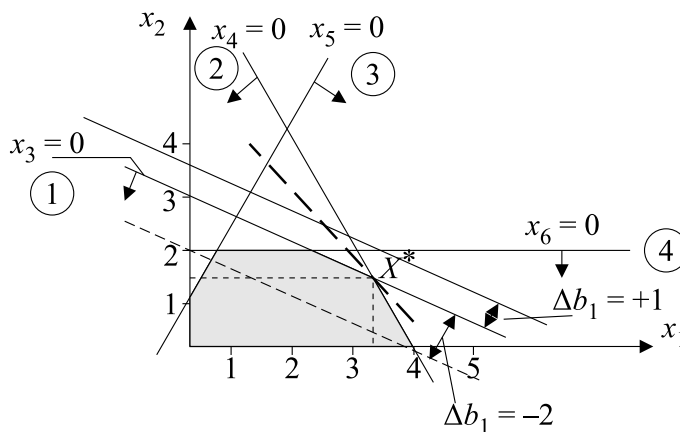


Рис. 2.13 – Графическое решение задачи

Решением прямой задачи является вектор X^* с переменными $x_1^* = 3\frac{1}{3}$; $x_2^* = 1\frac{1}{3}$; $x_3^* = 0$; $x_4^* = 0$; $x_5^* = 3$; $x_6^* = 2/3$. Решением двойственной задачи является вектор Y^* с переменными $y_1^* = 1/3$; $y_2^* = 4/3$; $y_3^* = 0$; $y_4^* = 0$; $y_5^* = 0$; $y_6^* = 0$. Дефицитными ресурсами являются ресурс 1 (остаточная переменная $x_3^* = 0$) и ресурс 2 ($x_4^* = 0$) с ценностью (степенью дефицитности) ресурса соответственно $y_1^* = 1/3$ и $y_2^* = 4/3$. Недефицитными ресурсами являются ресурс 3 ($x_5^* = 3$, $y_3^* = 0$) и ресурс 4 ($x_6^* = 2/3$, $y_4^* = 0$).

Анализ ограничений по ресурсам

Рассмотрим два случая анализа:

- а) ограничений, по которым нет резерва;
- б) ограничений, по которым есть резерв.

Для ресурсов 1 и 2 (*случай а*) изменение запаса ведет к изменению оптимального значения целевой функции. Из симплекс-таблицы значение целевой функции выражается уравнением:

$$Z = 12\frac{2}{3} - \left(\frac{1}{3} \cdot x_3 + \frac{4}{3} \cdot x_4 \right).$$

При увеличении $x_3 > 0$, а значит, при уменьшении запаса ресурса 1, $b'_1 = b_1 - x_3$, следует уменьшение целевой функции. При $x_3 = 1$, значение целевой функции Z уменьшится на величину $y_1 = 1/3$. И наоборот, при увеличении запаса ресурса (уменьшении x_3) Z увеличивается на величину $y_1 = 1/3$.

В каких пределах запас дефицитных ресурсов можно изменить, чтобы их статус дефицитности не изменился, т. е. чтобы они оставались по-прежнему дефицитными? Анализ проводится по каждому из ресурсов. Проведем по первому. Изменим первый ресурс $b_1 = 6$ на $b_1 + \Delta b_1$. При этом первое ограничение примет вид: $x_1 + 2x_2 + x_3 = 6 + \Delta b_1$. Решив задачу, в которой, в отличие от исходной, правая часть первого ограничения заменена на $6 + \Delta b_1$, построим последнюю симплексную таблицу 2.14.

Таблица 2.14 – Симплекс-таблица

Базис	x_1	x_2	x_3	x_4	x_5	x_6	Решение
z	0	0	$\frac{1}{3}$	$\frac{4}{3}$	0	0	$12\frac{2}{3} + \frac{1}{3}\Delta b_1$
x_2	0	1	$\frac{2}{3}$	$-\frac{1}{3}$	0	0	$\frac{4}{3} + \frac{2}{3}\Delta b_1$
x_1	1	0	$-\frac{1}{3}$	$\frac{2}{3}$	0	0	$\frac{10}{3} - \frac{1}{3}\Delta b_1$
x_5	0	0	-1	1	1	0	$3 - 1\Delta b_1$
x_6	0	0	$-\frac{2}{3}$	$\frac{1}{3}$	0	1	$\frac{2}{3} - \frac{2}{3}\Delta b_1$

Сравнение таблицы 2.13 с 2.14 показывает, что в последней таблице изменился только столбец свободных членов. Его элементы равны исходным значениям свободного члена, сложенным с произведением соответствующего элемента столбца дополнительной переменной x_3 на приращение Δb_1 .

Чтобы решение, представленное в таблице 2.14, было опорным (базисные переменные должны быть неотрицательными, и в базисе не появилась переменная x_3 , определяющая остаток ресурса 1), необходимо выполнение системы неравенств:

$$x_2 = 4/3 + 2/3\Delta b_1 \geq 0;$$

$$x_1 = 10/3 - 1/3\Delta b_1 \geq 0;$$

$$x_5 = 3 - \Delta b_1 \geq 0;$$

$$x_6 = 2/3 - 2/3\Delta b_1 \geq 0.$$

Решив систему неравенств, находим пределы изменения Δb_1 : $-2 \leq \Delta b_1 \leq 1$.

Пределы применения ресурса Δb_1 , при которых сохраняется структура оптимального плана, найдем по формулам $\min b_1 = b_1 + \min \Delta b_1 = 6 - 2 = 4$; $\max b_1 = b_1 + \max \Delta b_1 = 6 + 1 = 7$.

Следовательно, в пределах $4 \leq b_1 \leq 7$ сохраняется структура оптимального плана, т. е. по-прежнему следует выпускать продукцию x_1 и x_2 . Однако значения координат оптимального плана и целевая функция изменяются (рис. 2.13).

Аналогично поступают с анализом ресурса 2.

Ресурсы 3 и 4 (*случай б*) полностью не расходуются, т. е. дополнительные переменные x_5 и x_6 являются базисными. Проведя аналогичные рассуждения как и в первом случае, например для ресурса 4 (он связан с переменной x_6), можно показать, что решение будет оставаться опорным оптимальным при условии:

$$x_6 = 2/3 + 1 \cdot \Delta b_4 \geq 0.$$

Отсюда: $-2/3 \leq \Delta b_4 < \infty$. При изменении ресурса b_4 в пределах $(2 - 2/3) \leq b_4 < \infty$ сохраняется статус ресурса.

Анализ коэффициентов целевой функции

Рассмотрим два случая анализа:

- а) при базисных переменных;
- б) при небазисных.

В вышеприведенном примере (*случай а*) базисными переменными служат x_1 , x_2 , x_5 и x_6 . Анализ делается по каждой из переменных. Сделаем анализ изменения коэффициента C_1 .

В целевой функции задачи дадим коэффициенту $C_1 = 3$ приращение ΔC_1 . Тогда целевая функция примет вид: $Z = (3 + \Delta C_1)x_1 + 2x_2$. Проведя симплексные преобразования с целевой функцией, придем к последней симплексной таблице, в которой строка целевой функции Z будет отличаться от строки Z в таблице 2.13 только по строке Z и будет иметь вид:

Базис	x_1	x_2	x_3	x_4	x_5	x_6	Решение
z	0	0	$\frac{1}{3} - \frac{1}{3} \cdot \Delta C_1$	$\frac{4}{3} + \frac{2}{3} \cdot \Delta C_1$	0	0	$12\frac{2}{3} + \frac{10}{3} \cdot \Delta C_1$

Отличие в строке Z наблюдается в коэффициентах при небазисных переменных на слагаемую величину, составленную из произведения соответствующих коэффициентов строки базисной переменной x_1 и приращения ΔC_1 .

Оптимальные значения переменных будут оставаться переменными при значениях ΔC_1 , удовлетворяющих условию неотрицательности всех коэффициентов при небазисных переменных в Z -уравнении. Таким образом, должны выполняться следующие неравенства:

$$1/3 - 1/3\Delta C_1 \geq 0;$$

$$4/3 + 2/3\Delta C_1 \geq 0.$$

Отсюда $-2 \leq \Delta C_1 < 1$.

Пределы изменения коэффициента C_1 равны:

$$\min C_1 = C_1 + \min \Delta C_1 = 3 + 1 = 4; \max C_1 = C_1 + \max \Delta C_1 = 3 + 1 = 4.$$

Целевая функция при этом будет изменяться в соответствии с выражением

$$12\frac{2}{3} + \frac{10}{3} \cdot \Delta C_1.$$

Итак, при изменении C_1 в пределах $[1; 4]$ сохраняется структура и числовые значения оптимального плана. Однако изменяется значение целевой функции и двойственных оценок.

Любое изменение коэффициента при небазисной переменной (*случай б*), например C_3 , в целевой функции $z = 3x_1 + 2x_2 + (0 + \Delta C_3)x_3 + 0 \cdot x_5 + 0 \cdot x_6$ приводит лишь к тому, что в заключительной симплекс-таблице изменяется только этот коэффициент в строке Z , которая примет вид:

Базис	x_1	x_2	x_3	x_4	x_5	x_6	Решение
z	0	0	$\frac{1}{3} - \Delta C_3$	$\frac{4}{3}$	0	0	$12\frac{2}{3}$

Видно, что единственное отличие от Z -уравнения до введения ΔC_3 состоит в том, что коэффициент при x_3 уменьшился на ΔC_3 . Тогда для сохранения структуры плана должно выполняться условие:

$$y_1 - \Delta C_3 = 1/3 - \Delta C_3 \geq 0.$$

Отсюда $-\infty < \Delta C_3 \leq 1/3$. А так как $C_3 \geq 0$, то все характеристики оптимального плана (кроме y_1) будут неизменными в пределах $0 \leq C_3 \leq 1/3$.

2.5 Дробно-линейное программирование

Рассмотрим следующую задачу.

Завод выпускает продукцию n видов $p_1, \dots, p_n$. В процессе производства используются m видов сырья $S_1, \dots, S_m$, запасы которого ограничены. Требуется составить производственный план таким образом, чтобы обеспечить максимально возможную рентабельность работы заводов. Нормы расхода сырья, его запасы, а также удельные и условно постоянные затраты и прибыль, получаемая заводом от реализации одного изделия, приведены в таблице 2.15.

Условно-переменные (удельные) затраты — затраты, которые изменяются прямо пропорционально объемам выпуска товаров. Условно-постоянные затраты — это затраты, которые практически не зависят от изменения количества выпускаемой продукции (затраты на освещение, арендная плата и др.).

Пусть $x_1, \dots, x_n$ — объем выпуска продукции $p_1, \dots, p_n$. Математическая модель будет иметь вид:

$$Z(X) = \frac{c_1x_1 + \dots + c_nx_n}{d_1x_1 + \dots + d_nx_n + d_0} = \frac{\sum_{j=1}^n c_jx_j}{\sum_{j=1}^n d_jx_j + d_0} \rightarrow \max;$$

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

$$x_j \geq 0, \quad j = \overline{1, n}.$$

Таблица 2.15 – Исходные данные

Сырье	Продукция					Запасы сырья
	p_1	...	p_j	...	p_n	
...	...	...	...	...	...	...
S_i	a_{i1}	...	a_{ij}	...	a_{in}	b_i
...	...	...	...	...	...	...
Удельные затраты на 1 изделие	d_1	...	d_j	...	d_n	Условно-постоянные затраты d_0
Прибыль от реализации изделия	C_1	...	C_j	...	C_n	

Сведем данную задачу к задаче линейного программирования. Обозначим через $1/\delta$ знаменатель целевой функции, $\delta > 0$:

$$\sum_{j=1}^n d_j x_j + d_0 = \frac{1}{\delta}.$$

Умножим правую и левую части ограничения модели на δ . Получим:

$$Z = \sum_{j=1}^n c_j x_j \delta \rightarrow \max;$$

$$\sum_{j=1}^n a_{ij} x_j \delta \leq b_i \delta, \quad i = \overline{1, m};$$

$$\sum_{j=1}^n d_j x_j \delta + d_0 \delta = 1;$$

$$x_j \geq 0, \quad \delta > 0, \quad j = \overline{1, n}.$$

Обозначим $y_j = x_j \delta, j = \overline{1, n}$. Получим модель линейного программирования с переменными $x_j, j = \overline{1, n}$ и δ :

$$Z = \sum_{j=1}^n C_j y_j \rightarrow \max;$$

$$\sum_{j=1}^n a_{ij} y_j - b_i \delta \leq 0, \quad i = \overline{1, m};$$

$$\sum_{j=1}^n d_j y_j + d_0 \delta = 1;$$

$$y_j \geq 0, \quad j = \overline{1, n}, \quad \delta > 0.$$



Пример 2.4

$$Z = \frac{0,012x_1 + 0,008x_2}{0,01x_1 + 0,04x_2 + 1} \rightarrow \max;$$

$$1,8x_1 + 0,2x_2 \leq 20; \quad (*)$$

$$2,558x_1 + 1,2x_2 \leq 45; \quad (**)$$

$$x_1, x_2 \geq 0.$$

Решение:

$$0,01x_1 + 0,04x_2 + 1 = \frac{1}{\delta};$$

$$Z = 0,012x_1\delta + 0,008x_2\delta \rightarrow \max;$$

$$1,8x_1\delta + 0,2x_2\delta \leq 20\delta;$$

$$2,55x_1\delta + 1,2x_2\delta \leq 45\delta;$$

$$0,01x_1\delta + 0,04x_2\delta + \delta = 1;$$

$$x_1 \geq 0, x_2 \geq 0, \delta > 0.$$

$$y_1 = x_1\delta; y_2 = x_2\delta;$$

$$1,8y_1 + 0,2y_2 - 20\delta \leq 0;$$

$$2,55y_1 + 1,2y_2 - 45\delta \leq 0;$$

$$0,1y_1 + 0,04y_2 + \delta = 1;$$

$$y_1 \geq 0, y_2 \geq 0, \delta > 0.$$

Решаем симплекс-методом. Получим $y_1 = 5, y_2 = 10, z = 14\%$.

$$\delta = 1 - 0,01y_1 - 0,04y_2 = 0,55.$$

Тогда $x_1 = 5/0,55 = 9,09; x_2 = 10/0,55 = 18,18$.

Графическое решение задачи приведено на рисунке 2.14.

Ограничения задачи — это множество допустимых решений, лежащих в четырехугольнике $ABCO$. Целевая функция задачи — линии уровня цели, проходящие через точку S . Координаты точки S определяются в результате решения системы уравнений:

$$\begin{cases} 0,012x_1 + 0,008x_2 = 0 & (\text{числитель целевой функции}); \\ 0,01x_1 + 0,04x_2 + 1 = 0 & (\text{знаменатель целевой функции}). \end{cases}$$

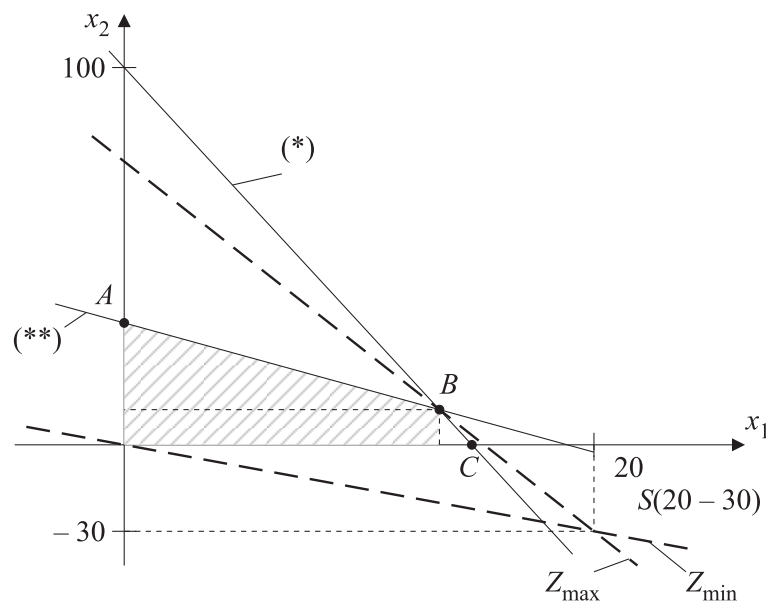


Рис. 2.14 – Геометрическая интерпретация задачи



Контрольные вопросы по главе 2

1. Что такое линейное программирование (ЛП)?
2. Дайте геометрическую интерпретацию постановки задачи линейного программирования.
3. Дайте определение основной теоремы линейного программирования.
4. Сколько базисных переменных находится в решении задач ЛП?
5. Дайте описание графического способа решения задач ЛП с двумя и многими переменными.
6. В чем заключается идея симплекс-метода?
7. В чем отличие прямого, двойственного и двухэтапного симплекс-алгоритмов?
8. Как по оптимальной симплекс-таблице:
 - а) определить решение прямой и обратной (двойственной) задачи ЛП?
 - б) оценить единственность решения задачи ЛП?
 - в) сделать анализ модели на чувствительность изменения коэффициентов в целевой функции и ограничений по ресурсам?
9. Дайте определения теорем двойственности для ЗЛП.
10. Дайте экономическую интерпретацию теорем двойственности.

Глава 3

ЗАДАЧИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ ТРАНСПОРТНОГО ТИПА

3.1 Транспортная задача линейного программирования

3.1.1 Математическая модель задачи

Содержательная и математическая постановки транспортной задачи были рассмотрены в разделе 2.1.2. Напомним ее в более общем случае. Пусть имеется m ($i = \overline{1, m}$) поставщиков, располагающих некоторым однородным продуктом в объемах по a_i единиц, и n ($j = \overline{1, n}$) получателей с объемами потребления по b_j единиц. Задана матрица $C = \|c_{ij}\|$, где c_{ij} — стоимость перевозки единицы продукции от i -го поставщика j -му потребителю. Возникает задача определения плана перевозок $X = \|x_{ij}\|$, где x_{ij} — количество единиц продукции, поставляемой по коммуникации (i, j) , который бы минимизировал суммарную стоимость всех перевозок. Математическая модель задачи будет иметь вид:

$$\min: Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij},$$

при ограничениях:

$$\begin{aligned} \sum_{j=1}^n x_{ij} &\leq a_i, \quad i = \overline{1, m}; \\ \sum_{i=1}^m x_{ij} &\geq b_j, \quad j = \overline{1, n}; \\ x_{ij} &\geq 0, \quad i = \overline{1, m}; \quad j = \overline{1, n}. \end{aligned}$$

Если $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = \sum_{i=1}^m \sum_{j=1}^n x_{ij}$, то задача называется закрытой (сбалансированной), в противном случае — открытой.

Переход к закрытой форме задачи. В случае если запасы поставщиков больше потребностей потребителей $\sum_{i=1}^m a_i > \sum_{j=1}^n b_j$, вводят $(n+1)$ -го фиктивного потребителя, запросы которого равны излишку запаса, т. е.

$$b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j.$$

Тарифы c_{in+1} считаются равными нулю. Получим расширенную закрытую задачу. Поставки x_{in+1} в оптимальном плане расширенной задачи покажут остатки продукции на складах поставщиков.

Если потребности превышают запасы $\sum_{i=1}^m a_i < \sum_{j=1}^n b_j$, то вводят $(m+1)$ -го фиктивного поставщика. Его запасы считают равными недостающей продукции:

$$a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i.$$

Тарифы c_{m+1j} равны некоторому большому положительному числу. В расширенной задаче получим баланс потребностей и запасов. Поставки x_{m+1j} в оптимальном плане расширенной задачи покажут объемы недовыполнения заказов.

Транспортная задача может быть решена симплекс-методом. Благодаря специфике системы ограничений разработаны методы поиска решения на матрице допустимых решений, что значительно упрощает процедуру симплексного метода. Специфика ограничений сбалансированной задачи:

- коэффициенты при x_{ij} равны единице;
- каждая переменная x_{ij} встречается дважды;
- в случае баланса всегда имеется допустимое решение задачи;
- система ограничений линейна, зависима, число линейно независимых уравнений равно $m+n+1$, а следовательно, число базисных переменных в транспортной задаче тоже равно $m+n+1$, где $(m+n)$ — число ограничений сбалансированной задачи.

При любом методе решения транспортной задачи начинают с нахождения начального опорного плана [1–3].

3.1.2 Определение начального опорного плана задачи

Способ северо-западного угла

Заполнение матрицы решения (перевозок) начинается каждый раз с верхнего левого угла. В таблице 3.1 приведен начальный опорный план для задачи размерностью (2×2) , взятой из раздела 2.1.2. Слева таблицы построчно отражаются объемы поставщиков a_i , сверху таблицы — объемы поставщиков b_j . В клетку (i, j) всегда проставляется значение переменной $x_{ij} = \min(a_i, b_j)$ с соответствующей корректировкой объемов a_i и b_j .

Таблица 3.1 – Опорный план 1

	$b_1 = 42$	$b_2 = 58$
$a_1 = 50$	42	8
$a_2 = 50$	–	50

Прочерк в таблице означает, что переменная $x_{21} = 0$ не вошла в базис. Число заполненных клеток (базисных переменных) равно $m + n - 1 = 3$, что соответствует угловой точке симплекса.

В случае если $a_i = b_j$, опорный план будет вырожденным, т. е. на следующем шаге величина поставки записывается равной нулю (в базисе будет переменная, равная нулю). Недостаток данного способа: он не учитывает матрицы тарифов. Поэтому опорный план может оказаться далеким от оптимального плана.

Способ минимального элемента

Заполнение матрицы решения начинается каждый раз с клетки с минимальным значением элемента c_{ij} . В таблице 3.2 приведен опорный план для задачи размерностью (2×2) с теми же объемами a_i и b_j , что и в примере, рассмотренном выше. В углах клеток проставлены оценки c_{ij} .

Таблица 3.2 – Опорный план 2

	42	58
50	– ²	50 ¹
50	42 ²	8 ³

Значение целевой функции для опорного плана, найденного способом северо-западного угла, равно $Z^0 = 2 \cdot 42 + 1 \cdot 8 + 2 \cdot 0 + 3 \cdot 50 = 242$.

Значение целевой функции для опорного плана, найденного способом минимального элемента, равно $Z' = 2 \cdot 0 + 1 \cdot 50 + 2 \cdot 42 + 3 \cdot 8 = 158$.

Способ У. Фогеля

Дает опорный план, близкий к оптимальному. Его используют как приближенный метод решения транспортной задачи. Заполнение матрицы решений начинается каждый раз с клетки (i_0, j_0) , которая определяется следующим образом. Для каждой строки и столбца определяются оценки sa_i и sb_j как разность между двумя наименьшими оценками c_{ij} , соответственно каждой строки и столбца:

$$sa_i = \min_{j \neq k} c_{ij} - c_{ik}, \text{ где } c_{ik} = \min_j c_{ij},$$

$$sb_j = \min_{i \neq k} c_{ij} - c_{kj}, \text{ где } c_{kj} = \min_i c_{ij}.$$

Далее находим строку i_0 или столбец j_0 с наибольшей оценкой sa_i и sb_j . В ней (нём) клетку (i_0, j_0) с наименьшим тарифом c_{ij} . Далее, как и ранее, заполняется клетка $x_{i_0 j_0} = \min(a_{i_0}, b_{j_0})$ с соответствующей корректировкой объемов a_{i_0} и b_{j_0} .

3.1.3 Распределительный метод

Обратимся к примеру, рассмотренному в разделе 2.1.2. Определим начальный опорный план X^0 способом северо-западного угла (табл. 3.3) $Z^0(X^0) = 242$ тыс. руб.

Таблица 3.3 – Схема сдвига

	42	58
50	– 42	2 1
50	2 +	3 –

Другой угловой точкой множества допустимых решений задачи может быть матрица X^1 , в которой значение перевозки x_{21} будет отличное от нуля $x_{21} \neq 0$. Какую величину перевозки x_{21} надо установить, чтобы получить новую угловую точку, новое решение? Необходимо, чтобы в матрице было три заполненных клетки, так как для четырех заполненных клеток решение будет располагаться не в угловой точке, а внутри области допустимых решений.

Проведем такие рассуждения. Если по маршруту (2–1) повезем пиломатериал в объеме Q , т. е. $x_{21} = 0 + Q$ (поемим клетку (2, 1) знаком «+»), то с маршрута (2–2) этот объем надо снять, т. е. $x_{22} = x_{22} - Q$, чтобы $x_{21} + x_{22} = a_2$ или $(0 + Q) + (50 - Q) = 50$ (поемим клетку (2, 2) знаком «–»).

Далее аналогично, $x_{12} = x_{12} + Q$ (клетку (1, 2) поемим знаком «+»), $x_{11} = x_{11} - Q$ (клетку (1, 1) поемим знаком «–»).

Максимальный объем перераспределения Q может быть равен минимальному объему перевозок, клеточки которых помечены знаком «–», т. е. $Q = \min(50, 42) = 42$.

Новый план X^1 решения задачи представлен в таблице 3.4. Значение целевой функции для нового решения будет $Z^1 = 2 \cdot 0 + 50 \cdot 1 + 42 \cdot 2 + 8 \cdot 3 = 158$ тыс. руб.

Таблица 3.4 – Новый план

	42	58
50	– 42	2 1
50	2 42	3 8

Новое решение X^1 лучше, чем решение X^0 . Других решений (угловых точек) для данной задачи нет. А теперь сформулируем *условие оптимальности*, при котором достигается наилучшее решение.

Сравним две матрицы X^0 и X^1 и соответственно Z^0 и Z^1 . Для клетки, помеченной знаком «+», будем делать обозначения x_{ij}^+ и C_{ij}^+ , а для клетки со знаком «–» будем делать обозначения x_{ij}^- и C_{ij}^- .

$$Z^0 = C_{11}^- x_{11}^- + C_{12}^+ x_{12}^+ + C_{21}^+ x_{12}^+ + C_{22}^- x_{22}^-,$$

$$Z^1 = C_{11}^- (x_{11}^- - Q) + C_{12}^+ (x_{12}^+ + Q) + C_{21}^+ (x_{12}^+ + Q) + C_{22}^- (x_{22}^- - Q).$$

Вычтем из Z^0 выражение Z^1 :

$$Z^0 - Z^1 = \Delta Z = C_{11}^- Q - C_{12}^+ Q - C_{21}^+ Q + C_{22}^- Q = Q[(C_{11}^- + C_{22}^-) - (C_{12}^+ + C_{21}^+)] > 0,$$

$$\Delta Z = 42[(2 + 3) - (1 + 2)] = 42 \cdot 2 = 84; \quad Z^0 - Z^1 = 242 - 158 = 84.$$

Из анализа видно, что решение можно улучшить, если значение в квадратных скобках положительное, и, наоборот, решения лучше нет, если выражение в квадратных скобках меньше нуля или равно нулю. Ломаная линия в таблице 3.3, проходящая через одну небазисную (незаполненную) клетку и остальные базисные (заполненные) клетки, называется **контуром**. В любом допустимом плане транспортной задачи всегда можно провести через любую клетку контур, причем единственный (обозначим его буквой L). Если контур образует самопересекающаяся ломаная линия, то точки ее самопересечения вершин не образуют. Примеры контуров показаны на рисунке 3.1.

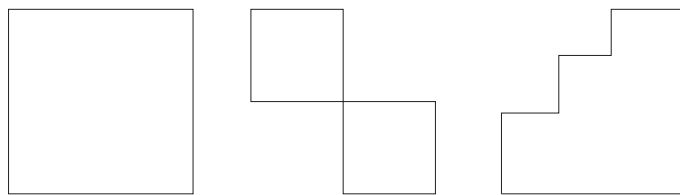


Рис. 3.1 – Примеры контуров

В плане перевозок всегда можно сделать перераспределение, тем самым перейти от одного решения к другому решению. Выражение в квадратных скобках называют **ценой цикла** для клетки $(2, 1)$. Запишем это выражение в общем виде для клетки (i, j) :

$$\Delta_{ij} = \sum_{i,j \in L} C_{ij}^- - \sum_{i,j \in L} C_{ij}^+.$$

Изменение ΔZ целевой функции, вызываемое сдвигом по циклу на величину Q , запишется:

$$\Delta Z = Q \Delta_{ij} = Q \left(\sum_{i,j \in L} C_{ij}^- - \sum_{i,j \in L} C_{ij}^+ \right),$$

где $\sum_{i,j \in L} C_{ij}^-$ и $\sum_{i,j \in L} C_{ij}^+$ — соответственно сумма тарифов по отрицательно и положительно означенным клеткам контура.

Алгоритм решения транспортной задачи распределительным методом выглядит следующим образом.

1. Строим начальный опорный план. Занятыми должны быть $(m+n-1)$ клеток.
2. Для небазисных клеток строим контуры, определяем цены циклов. Если все $\Delta_{ij} \leq 0$, то план оптимальный, иначе переходим к новому плану путем перераспределения через ту небазисную клетку, цена цикла которой имеет наибольшее значение, и на начало п. 2.

3.1.4 Метод потенциалов

Запишем сбалансированную транспортную задачу как прямую задачу линейного программирования. Для ограничений этой задачи введем двойственные оценки (потенциалы) на поставщиков U_i , $i = \overline{1, m}$ со знаком «минус», на потребителей продукции V_j , $j = \overline{1, n}$ и рядом запишем обратную ей двойственную задачу. Знак «минус» перед оценкой U_i , $i = \overline{1, m}$ оправдывается удобством экономической интерпретации: потенциал потребителя увеличивается при переводе единицы продукции на стоимость этой продукции.

$$V_j = U_i + C_{ij}.$$

Прямая задача	Оценки	Двойственная задача
$\min : Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$		$\max : Z = \sum_{j=1}^n v_j b_j - \sum_{i=1}^m u_i a_i$
$\sum_{j=1}^n x_{ij} = a_i, i = \overline{1, m}$	$-u_i$	$v_j - u_i \leq c_{ij}, i = \overline{1, m}; j = \overline{1, n}$
$\sum_{i=1}^m x_{ij} = b_j, j = \overline{1, n}$	v_j	v_j, u_i — любого знака
$x_{ij} \geq 0, i = \overline{1, m}; j = \overline{1, n}$		

На основании второй теоремы двойственности для оптимальных решений прямой $X^* = \|x_{ij}^*\|$ и двойственной $Y^* = (-u_i^*, v_j^*)$ задач должны выполняться условия $x_{ij}^*(v_j^* - u_i^* - c_{ij}) = 0, i = \overline{1, m}; j = \overline{1, n}$.

Если считать допустимый план X оптимальным, то для заполненных клеток i, j (базисных переменных x_{ij}) должны выполняться условия $\Delta_{ij} = v_j - u_i - c_{ij} = 0, i = \overline{1, m}; j = \overline{1, n}$. Так как число базисных переменных равно $(m + n - 1)$, а число приведенных равенств $(m + n)$, то оценки v_j и u_i могут быть определены, если значение одной из оценок выбрать произвольно (обычно берут $u_1 = 0$), а другие оценки вычисляются через тарифы c_{ij} базисных переменных.

Тогда, с другой стороны, для незаполненных клеток i, j ($x_{ij} = 0$) в оптимальном плане X оценки Δ_{ij} должны быть неположительными, т. е. $\Delta_{ij} = v_j - u_i - c_{ij} \leq 0, i = \overline{1, m}; j = \overline{1, n}$ (из ограничения двойственной задачи). Следовательно, признаком оптимальности плана следует считать отсутствие положительных оценок Δ_{ij} для незаполненных клеток. Отсюда алгоритм метода потенциалов состоит из следующих шагов:

- 1) строим начальный опорный план, заполненными должны быть $(m + n - 1)$ клеток;
- 2) предполагая, что план оптимальный, определяем потенциалы v_j, u_i через заполненные клетки $u_1 = 0; v_j - u_i - c_{ij} = 0, i = \overline{1, m}; j = \overline{1, n}$;
- 3) определяем оценки Δ_{ij} для незаполненных клеток, которые в оптимальном плане должны быть неположительными: $\Delta_{ij} = v_j - u_i - c_{ij} \leq 0, i = \overline{1, m}; j = \overline{1, n}$;
- 4) определяем клетку (i_0, j_0) с максимальной оценкой Δ_{ij} :

$$\max_{\Delta_{ij} > 0} \Delta_{ij} = \Delta_{i_0 j_0};$$

- 5) строим контур L через вновь вводимую в базис переменную $x_{i_0 j_0}$;

- б) переходим к другому допустимому плану задачи путем перераспределения объемов перевозок по контуру L и на шаг 2.

$$x'_{ij} = \begin{cases} x_{ij} & \text{— для клеток, не входящих в контур;} \\ x_{ij}^+ + Q & \text{— для положительно помеченных клеток контура } (i, j) \in L; \\ x_{ij}^- - Q & \text{— для отрицательно помеченных клеток контура } (i, j) \in L. \end{cases}$$



Пример 3.1

Минимизировать транспортные расходы (табл. 3.5).

Таблица 3.5 – Условия задачи

	100	120	130	150
200	4	6	7	2
150	2	4	3	1
150	9	7	4	2

В углах клеток проставлены тарифы c_{ij} , $i = \overline{1, 3}$; $j = \overline{1, 4}$.

Решение:

1. Строим начальный опорный план X^0 методом минимального элемента (табл. 3.6).

Таблица 3.6 – Опорный план

	100 –	120 20	130 –	150 –
200 100	100 4	100 6	– 7	– 2
150 –	0 2	– 4	– 3	150 1
150 20	– 9	20 7	130 4	– 2

Заполненных клеток должно быть $m + n - 1 = 3 + 4 - 1 = 6$. Опорный план вырожденный $X_{21} = 0$.

2. Определяем потенциалы через заполненные клетки (табл. 3.7).

Таблица 3.7 – Потенциалы поставщиков и потребителей

4	6			$u_1 = 0$
2			1	$u_2 = 2$
	7	4		$u_3 = -1$
$v_1 = 4$	$v_2 = 6$	$v_3 = 3$	$v_4 = 3$	

$$u_1 = 0; \quad u_i + c_{ij} = v_j, \quad i = \overline{1, 3}; \quad j = \overline{1, 4}.$$

Тарифы в таблице проставлены только для заполненных клеток.

3. Определяем оценки Δ_{ij} для незаполненных клеток (табл. 3.8).

Таблица 3.8 – Оценка свободных клеток

		-4 ⁷	+1 ²	$u_1 = 0$
	0 ⁴	-2 ³		$u_2 = 2$
-4 ⁹			+2 ²	$u_3 = -1$
$v_1 = 4$	$v_2 = 6$	$v_3 = 3$	$v_4 = 3$	

$$\Delta_{ij} = v_j - u_i - c_{ij}, \quad i = \overline{1, 3}; \quad j = \overline{1, 4}.$$

План не удовлетворяет условию оптимальности, т.к. есть положительные оценки Δ_{ij} .

4. Перспективная клетка для ввода в базис (3, 4).

5. Строим контур через клетку (3, 4) (табл. 3.9).

Таблица 3.9 – Контур клетки (3, 4)

-	+		
100	100		
+			-
0			150
	-		+
	20	130	

6. Переходим к новому допустимому плану (табл. 3.10).

$$Q = \min\{100; 150; 20\} = 20.$$

Таблица 3.10 – Новый план задачи

80	120		
20			130
		130	20

Из базиса выведена переменная $x'_{32} = 0$. В базисе осталось 6 переменных. Далее на шаг 2.

За 4 итерации можно получить *оптимальное* решение (табл. 3.11) с целевой $Z(X^*) = 1640$.

Таблица 3.11 – Оптимальное решение

	70 ⁶		130 ²
100 ²	50 ⁴		
		130 ⁴	20 ²

Если определить потенциалы u_i , v_j и оценки Δ_{ij} , можно увидеть, что все оценки для незаполненных клеток $\Delta_{ij} > 0$, а $\Delta_{11} = 0$. Это говорит о том, что план оптимальный, но не единственный.

Введем в базис клетку (1, 1), построим контур и сделаем сдвиг по нему на величину $Q = 70$. Получим новый оптимальный план (табл. 3.12) с тем же значением целевой функции $Z(X^*) = 1640$.

Таблица 3.12 – Оптимальное решение

70			130
30	120		
		130	20

3.2 Транспортная задача с промежуточными пунктами

Допустим, имеется $m(i = \overline{1, m})$ пунктов производства, $n(j = \overline{1, n})$ — пунктов потребления и $p(r = \overline{1, p})$ — промежуточных баз. Как и в обычной транспортной задаче, обозначим через a_i , b_j соответственно объемы поставок и потребления. Пусть d_r — мощность r -й базы, c_{ir} и c_{rj} — соответственно стоимость перевозки единицы продукции от поставщиков на базы и от баз к потребителям. Тогда модель задачи примет вид:

$$\min Z: = \sum_{i=1}^m \sum_{r=1}^p c_{ir} x_{ir} + \sum_{r=1}^p \sum_{j=1}^n c_{rj} x_{rj},$$

при ограничениях:

$$\begin{aligned} \sum_{r=1}^p x_{ir} &\leq a_i, \quad i = \overline{1, m}; \\ \sum_{i=1}^m x_{ir} &\leq d_r, \quad r = \overline{1, p}; \\ \sum_{r=1}^p x_{rj} &\geq b_j, \quad j = \overline{1, n}; \\ x_{ir} &\geq 0; \quad x_{rj} \geq 0. \end{aligned}$$

В случае полностью сбалансированной задачи, т. е. $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = \sum_{r=1}^p d_r$, схема перевозок от поставщиков до баз не оказывает влияние на схему перевозок от баз до потребителей. В таких условиях задачу можно решать по частям для каждого этапа перевозок: от поставщиков до баз и от баз до потребителей продукции.

В случае частично сбалансированной задачи, а именно:

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j < \sum_{r=1}^p d_r,$$

оптимальный план двухэтапной транспортной задачи отличен от плана, полученного объединением оптимальных планов решения транспортной задачи для каждого этапа в отдельности. В таких условиях двухэтапную задачу сводят к классической с размерностью $(m + p) \times (p + n)$. Матрица тарифов будет состоять из четырех блоков (табл. 3.13).

Таблица 3.13 – Объединенная матрица тарифов

	d_1	$\dots$	d_p	b_1	$\dots$	b_n
a_1				M	M	M
$\dots$		$\ c_{ir}\ $		M	M	M
a_m				M	M	M
d_1	0	M	M			
$\dots$	M	0	M		$\ c_{ir}\ $	
d_p	M	M	0			

В первом левом верхнем блоке будем отражать связи поставщиков с базами (ir), в четвертом — связи баз с потребителями (rj). Второй правый верхний блок показывает связи поставщиков с потребителями, а так как таких непосредственных перевозок нет, то в этом блоке все тарифы считаются равными M (где M — большое число). Третий левый нижний блок размерностью $(p \times p)$ отражает связи между базами, поэтому все тарифы также считаются равными M , кроме диагональных.

Диагональные тарифы, отражающие затраты на переезд внутри базы, равны нулю. Сама диагональ называется фиктивной, т. к. поставки, найденные в результате решения задачи, в этих клетках будут означать величину неиспользованной мощности базы.

Решение двухэтапной транспортной задачи на объединенной матрице тарифов начинается с определения допустимого начального плана. Вначале заполняется блок первый или четвертый любым из способов определения начального плана. Затем заполняется фиктивная диагональ, и только потом распределяются поставки в другом незаполненном блоке (четвертом или первом).

После определения общего допустимого плана применяют известные методы поиска наилучшего решения. При этом имеется особенность определения контура сдвига: если контур пересчета проходит через фиктивную диагональ, то он обязательно проходит через нее дважды: один раз с пометкой «плюс», другой раз — с пометкой «минус».

3.3 Задача о назначениях

3.3.1 Математическая модель задачи

Содержательная и математическая постановки задачи были рассмотрены в разделе 2.1.3. Напомним ее в более общем случае.

Пусть необходимо выполнить n ($i = \overline{1, n}$) работ. Для этого используется n ($j = \overline{1, n}$) исполнителей, каждый из которых в состоянии выполнить любую работу. Известны затраты (c_{ij}) на выполнение j -м исполнителем i -й работы. Требуется назначить каждого исполнителя на одну работу так, чтобы минимизировать суммарные затраты. Математическая постановка задачи будет иметь вид

$$\min: Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij},$$

при ограничениях:

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= 1, \quad i = \overline{1, n}; \\ \sum_{i=1}^n x_{ij} &= 1, \quad j = \overline{1, n}; \\ x_{ij} &= \begin{cases} 1, & \text{если на } i\text{-ю работу назначен } j\text{-й исполнитель;} \\ 0, & \text{в противном случае.} \end{cases} \end{aligned}$$

Ограничения первого вида означают, что на каждую i -ю работу должен быть назначен только один исполнитель. Ограничения второго вида означают, что каждый j -й исполнитель должен быть назначен для выполнения только одной работы.

Данная задача о назначениях относится к транспортной при условии $a_i = b_j = 1$. Как и в транспортной, можно говорить об открытой (несбалансированной) задаче о назначениях, если число работ отлично от числа исполнителей. Если число работ меньше числа исполнителей ($m < n$), вводят $n - m$ фиктивных работ (столбцы) с нулевыми коэффициентами матрицы затрат, $c_{im+1} = 0, \dots, c_{in} = 0, i = \overline{1, n}$. В случае же ($m > n$) вводят $m - n$ фиктивных исполнителей (строки), которые не в состоянии выполнить предлагаемые работы, а значит, соответствующие элементы c_{ij} матрицы затрат можно полагать очень большими (M).

Если задача о назначениях ставится при условии получения максимума эффективности, то ее сводят к задаче на минимум следующим образом. Пусть дана матрица эффективностей $C = \|c_{ij}\|$. В каждом столбце найдем максимальный элемент l_j , т. е. $l_j = \max_i c_{ij}$. Перейдем к матрице $C' = \|c'_{ij}\| = \|l_j - c_{ij}\|$. Задача о назначениях с матрицей эффективности C эквивалентна задаче о назначениях с матрицей затрат (потерь) C' . В самом деле, если первая задача — это $\max_{X \in \Omega} Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$,

а вторая — $\max_{X \in \Omega} Z' = \sum_{i=1}^n \sum_{j=1}^n c'_{ij} x_{ij}$, где Ω — область допустимых решений задачи, то

$$Z' = \sum_{i=1}^n \sum_{j=1}^n (l_j - c_{ij}) x_{ij} = \sum_{j=1}^n l_j \sum_{i=1}^n x_{ij} - \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} = \sum_{j=1}^n l_j - Z, \quad \text{т. к.} \quad \sum_{i=1}^n x_{ij} = 1.$$

Очевидно, Z' достигает минимума при условии, что Z достигает максимума, $X \in \Omega$. Если по каким-либо причинам запрещается выполнение какой-либо работы каким-либо исполнителем, то в матрицу потерь вводят очень большую стоимость M .

Решение задачи о назначениях может быть найдено методами решения транспортной задачи. Но в силу особенности задачи о назначениях, заключающейся в том, что любое допустимое решение задачи является вырожденным (число ненулевых базисных переменных равно n), разработаны более эффективные методы ее решения. Все они базируются на двух простых утверждениях [1, 2, 8]:

- 1) решение задачи не изменится, если к любому столбцу или строке матрицы потерь прибавить (или вычесть) некоторую постоянную величину, т. е. если план X^* — оптимальный план задачи, то он также оптимален для функции цели Z' с матрицей $C' = \|c'_{ij}\|$, где $C'_{ij} = c_{ij} \pm u_i \pm v_j$, $i = \overline{1, n}$, $j = \overline{1, n}$;
- 2) если все $c_{ij} \geq 0$ и найден план X^* , такой, что $\sum_{i=1}^n \sum_{j=1}^n c_{ij}x_{ij}^*$, то X^* — оптимальный план.

Первое утверждение можно доказать так:

$$Z' = \sum_i \sum_j (c_{ij} \pm u_i \pm v_j) x_{ij} = \sum_i \sum_j c_{ij} x_{ij} \pm \sum_i u_i \sum_j x_{ij} \pm \sum_j v_j \sum_i x_{ij} = Z \pm \text{const.}$$

Отсюда следует, что минимизация исходной целевой функции Z приводит к такому же решению, как минимизация Z' .

Второе утверждение говорит о том, что в матрице потерь (исходной либо преобразованной по условию утверждения 1) для соответствующих элементов назначения ($x_{ij}^* = 1$) должны стоять нули: $c_{ij} = 0$. Данная совокупность нулевых элементов получила название системы независимых нулей. В строке и в столбце матрицы C не может быть более одного независимого нуля. Поэтому все алгоритмы поиска решения задачи о назначениях заключаются в преобразовании матрицы затрат $\|c_{ij}\|$ в матрицу с нулевыми элементами, образующими систему из n независимых нулей.

3.3.2 Венгерский метод решения задачи

Алгоритм метода состоит из предварительного и общего шагов, включающих не более чем $(n - 2)$ последовательно повторяющихся операций.

1. Подготовительный шаг:

- п. 1. Сводим задачу на $\min Z$ (если она была на $\max$) вычитанием из максимального элемента каждого столбца элементов c_{ij} ;
- п. 2. Вычитаем минимальные элементы в каждой из строк, а затем и столбцов матрицы затрат;
- п. 3. Выделяем (добавляем) систему независимых нулей (метим нули в столбцах (*), но чтобы не было в одной строке двух нулей с (*). Столбцы отмечают (+) — все элементы столбца называются выделенными. Пусть k — число независимых нулей.

2. Общий шаг:

Блок-схема алгоритма общего шага представлена на рисунке 3.2.

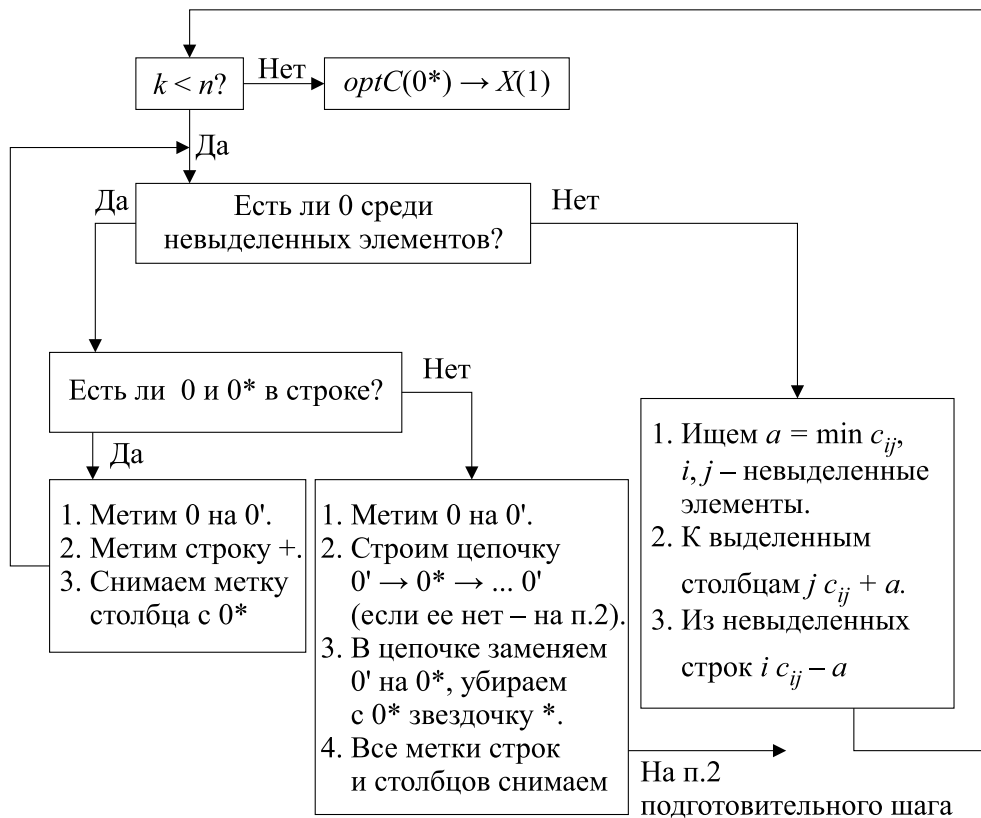
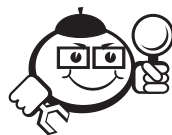


Рис. 3.2 – Блок-схема алгоритма общего шага



Пример 3.2

Решить задачу о назначениях с матрицей эффективности C :

$$C = \|c_{ij}\| = \begin{vmatrix} 10 & 7 & 6 & 9 \\ 3 & 7 & 5 & 1 \\ 5 & 6 & 8 & 3 \\ 6 & 11 & 7 & 5 \end{vmatrix}.$$

Решение:

Подготовительный шаг.

1.

10	7	6	9
3	7	5	1
5	6	8	3
6	11	7	5

max

10

11

8

9

2.

0	4	2	0
7	4	3	8
5	5	0	6
4	0	1	4

min

3.

+	+	+	
0*	4	2	0
4	1	0*	5
5	5	0	6
4	0*	1	4

Число независимых нулей $k = 3$. Переходим к общему шагу.

4. $\begin{array}{c|c|c|c} \oplus & + & + & \\ \hline 0^* & 4 & 2 & 0' \\ \hline 4 & 1 & 0^* & 5 \\ \hline 5 & 5 & 0 & 6 \\ \hline 4 & 0^* & 1 & 4 \end{array} +$ $c_{11} = 0^*$. Это означает, что первая работа поручается первому исполнителю. Однако она с таким же успехом может быть поручена и четвертому исполнителю, т. к. $c_{14} = 0$.

Поэтому данную оговорку фиксируем плюсом в первой строке (для первой работы), снятием плюса в первом столбце (с первого исполнителя) и штрихом 0' над нулем элемента $c_{14} = 0'$ (возможное поручение четвертому исполнителю). Элементы матрицы в первом столбце станут невыделенными, а элементы матрицы в первой строке станут выделенными. Таким образом, выделенными элементами будут элементы в первой строке, во втором и третьем столбцах.

5. $\begin{array}{c|c|c|c} & + & + & \\ \hline 0^* & 4 & 2 & 0' \\ \hline 4 & 1 & 0^* & 5 \\ \hline 5 & 5 & 0 & 6 \\ \hline 4 & 0^* & 1 & 4 \end{array} +$ Данная матрица получена в результате прибавления минимального элемента среди невыделенных элементов ($c_{21} = c_{41} = c_{44} = 4$) к элементам выделенных столбцов и вычитанием его из невыделенных строк.

Получим новые невыделенные нулевые элементы $c_{21} = c_{41} = c_{44} = 0$.

6. $\begin{array}{c|c|c|c} \oplus & \oplus & & \\ \hline 0^* & 8 & 6 & 0' \\ \hline 0' & 1 & 0^* & 1 \\ \hline 1 & 5 & 0 & 2 \\ \hline 0' & 0^* & 1 & 0 \end{array} +$ Отмечаем возможные новые перепоручения $c_{21} = 0'$ и $c_{41} = 0'$. Других возможных перепоручений нет, но есть $c_{33} = 0$.

7. $\begin{array}{c|c|c|c|c} & & & & \\ \hline 0^* & 8 & 6 & 0' & + \\ \hline 0' & 1 & 0^* & 1 & + \\ \hline 1 & 5 & 0 & 2 & \\ \hline 0' & 0^* & 1 & 0 & + \end{array} +$ Метим штрихом $c_{33} = 0'$. Строим цепочку $c_{33} \rightarrow c_{23} \rightarrow c_{21} \rightarrow c_{11} \rightarrow c_{14}$. На цепочке нули со штрихом заменяем на нули со звездочкой.

Звездочки, находящиеся на цепочке, сохраняем. Все остальные выделения уничтожаем. Получаем новую матрицу 8, в которой число независимых нулей увеличивается на единицу.

8.

0	8	6	0*
0*	1	0	1
1	5	0*	2
0	0*	1	0

9.

$$X^* =$$

0	0	0	1
1	0	0	0
0	0	1	0
0	1	0	0

Число независимых нулей равно 4. Задача решена. Оптимальный план представлен в матрице 9. Значение целевой $Z = 9 + 3 + 8 + 11 = 31$.

.....

3.3.3 Метод минимальных линий

Алгоритм метода состоит из следующих шагов:

- 1) сводим задачу на $\min Z'$ (если она была на $\max$);
- 2) вычитаем минимальные элементы в каждой из строк, а затем и столбцов матрицы затрат;
- 3) вычеркиваем все нулевые элементы матрицы минимальным числом прямых линий. Если минимальное число линий будет равно размерности матрицы, то независимые нули в матрице существуют, идем на шаг 5. Иначе — на шаг 4;
- 4) выбирается наименьший элемент из невычеркнутых элементов. Его значение вычитается из каждого невычеркнутого элемента и прибавляется к каждому элементу, стоящему на пересечении проведенных прямых. Получаем нулевые элементы. Линии убираем и идем на шаг 3;
- 5) строим систему независимых нулей и находим решение.

.....



Пример 3.3

.....

Решим задачу о назначениях с матрицей эффективности, рассмотренной в примере выше.

Решение:

На шаге 2 получим матрицу с нулевыми элементами, а на шаге 3 вычеркиваем эти нулевые элементы минимальным числом линий. Надо отметить, что минимальное число линий можно провести различными вариантами. Минимальный элемент среди невычеркнутых элементов равен 1. Переходим к матрице 4, вычеркиваем нули минимальным числом линий (их опять три), переходим к матрице 5. Вычеркиваем нули, число линий равно 4. Далее ищем независимые нули (матрица 6) и решения задачи (матрица 7).

2.

0	4	2	0
4	1	0	5
5	5	0	6
4	0	1	4

3.

0	4	2	0
4	1	0	5
5	5	0	6
4	0	1	4

4.

0	4	3	0
3	0	0	4
4	4	0	5
4	0	2	4

5.

0	7	6	0
0	0	0	1
1	4	0	2
1	0	2	1

6.

0	4	2	0*
0*	4	3	8
5	5	0*	6
4	0*	1	4

7.

$X^* =$

0	0	0	1
1	0	0	0
0	0	1	0
0	1	0	0

3.4 Транспортные сети

3.4.1 Примеры сетевых транспортных задач

В предыдущих разделах рассматривались транспортные задачи, в которых требовалось составить планы перевозок продукции из исходных пунктов в пункты назначения. Транспортная задача (и ее варианты) — лишь одна из разновидностей задач, которые можно сформулировать и решить с помощью сетевых моделей. Рассмотрим следующие примеры:

- 1) проектирование кабельной сети для телевизионного обслуживания районов-новостроек. Следует выбрать проект, для реализации которого потребуется кабель минимальной длины для связи (прямой или через другие пункты) всех районов с телевизионным центром;
- 2) определение кратчайшего пути между двумя городами, проходящего по существующей сети шоссейных дорог;
- 3) определение максимальной пропускной способности водопровода от водозабора до потребителей воды.

Перечисленные сетевые задачи можно описать следующими типами моделей, имеющих широкий круг практических приложений:

- минимизации сети (пример 1);
- нахождения кратчайшего маршрута (пример 2);
- определения максимального потока.

Вышеупомянутые задачи можно сформулировать и в принципе решить как задачи линейного программирования. Однако из-за огромного числа переменных и ограничений сетевых задач непосредственное применение симплекс-метода нецелесообразно. Поэтому разработаны в силу их особенностей более эффективные алгоритмы.

Дальнейшее обобщение подобных задач ведет к понятию транспортной сети. Прежде всего, напомним некоторые определения из теории графов [27].

Говорят, что задан граф $G(I, U)$, если задано непустое множество вершин I и множество дуг U . Причем каждый элемент $u_{ij} \in U$ есть упорядоченная пара (i, j) элементов множества I . Если оба множества I и U конечны, то граф конечен. Геометрически множество I задается точками, а множество U — отрезками (дугами), соединяющими эти точки.

Путем L_{ij} (цепью) в графе называется последовательность сцепленных дуг, позволяющих перейти из одной i -й вершины в другую j -ю вершину. Путь, у которого начальная вершина совпадает с конечной, называется контуром или циклом. Если элементам графа поставить в соответствие некоторые параметры, то получим сеть. Параметр, заданный вершине, называют весом или интенсивностью вершины, а параметр, заданный дуге графа, — длиной или пропускной способностью дуги. Вершины графа, для которых интенсивность положительная, именуется источниками, а вершины, для которых интенсивность отрицательная, — стоками. Если же интенсивность вершины равна нулю, то вершина нейтральна. Пропускные способности дуг характеризуют максимальное количество груза, которое соответствующая коммуникация может пропустить за единицу времени.

3.4.2 Минимизация сети

Задача минимизации сети состоит в нахождении дуг, соединяющих вершины сети (т. е. каждая пара вершин соединена цепью) и имеющих минимальную суммарную длину. Очевидно, что решение задачи не должно содержать циклов. Например, дан полный граф из трех вершин (рис. 3.3). На дугах, соединяющих вершины 1, 2, 3, указаны их длины. Необходимо получить сеть, объединяющую все вершины с минимальной суммарной длиной последовательности дуг.

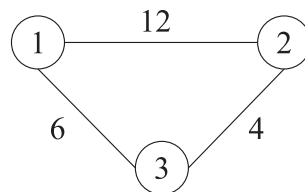


Рис. 3.3 – Транспортная сеть

Ясно, что в минимальную сеть войдут дуги (1, 3) и (3, 2) с общей длиной, равной 10 (рис. 3.4). Если соединить вершины 1 и 2, возникнет цикл, и получающаяся сеть уже не будет минимальной.



Рис. 3.4 – Минимальная сеть

В целом, задача минимизации сети состоит в том, чтобы из исходного графа $G(I, U)$ получить новый граф без циклов (так называемое дерево-остов) $GO(I, UO)$, такой, что

$$\sum_{U_{kr} \in UO} U_{kr} = \min_U \sum_{U_{kr} \in U} U_{kr}, \quad U_{kr} \in L_{ij}, \quad \forall i, j \in I.$$

Одним из самых простых алгоритмов построения требуемого графа является метод, предложенный польским математиком Г. Штейнгаузом [27]. Выбрав любую вершину, ее следует соединить с **ближайшим соседом**, а затем то же самое сделать со всеми остальными вершинами. Если образовалась сеть, охватывающая все вершины, то задача решена. Если вместо единого дерева получился лес не связанных друг с другом деревьев, то необходимо связать эти деревья через ближайшие вершины.

Формально алгоритм можно представить в виде следующих шагов:

- 1) множество связанных вершин $C = O$. Множество несвязных вершин $\bar{C} = I$. Выбираем $i \in \bar{C}$;
- 2) $C = C + i$; $\bar{C} = \bar{C} - i$. Если $C = I$, то решение найдено. Иначе, на шаг 3;
- 3) ищем вершину $k \in \bar{C}$, такую, что $U_{ik} = \min_{j \in \bar{C}} U_{ij}$, $i \in C$;
- 4) $i = k$ и на шаг 2.



Пример 3.4

Проложить сеть дорог наименьшей протяженностью (рис. 3.5). Отсутствие дуги между двумя вершинами означает, что соединение соответствующих пунктов связано с очень большими затратами либо физически невозможно.

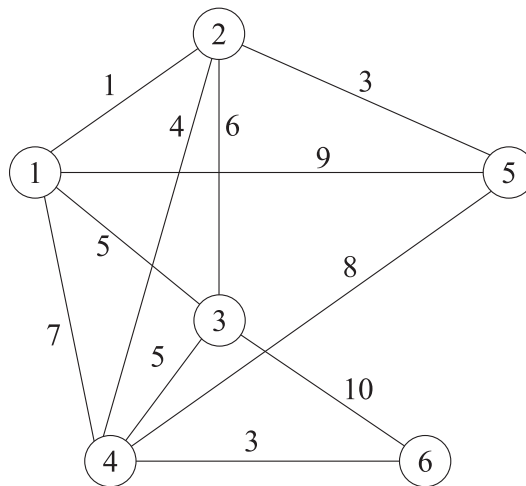


Рис. 3.5 – Транспортная (возможная) сеть

Решение:

- 1) $C = \{1\}$; $\bar{C} = \{2, 3, 4, 5, 6\}$;
- 2) $\min_{j \in \bar{C}} U_{1j} = \boxed{U_{12} = 1}$;
- 3) $C = \{1, 2\}$; $\bar{C} = \{3, 4, 5, 6\}$;

- 4) $\min_{j \in \bar{C}} U_{1j} = U_{13} = 5; \min_{j \in \bar{C}} U_{2j} = U_{25} = 3;$
- 5) $C = \{1, 2, 5\}; \bar{C} = \{3, 4, 6\};$
- 6) $\min_{j \in \bar{C}} U_{1j} = U_{13} = 5; \min_{j \in \bar{C}} U_{2j} = U_{24} = 4;$
- 7) $C = \{1, 2, 4, 5\}; \bar{C} = \{3, 6\};$
- 8) $\min(U_{13} = 5; U_{23} = 6; \min(U_{43} = 5; U_{46} = 3)) = U_{46} = 3;$
- 9) $C = \{1, 2, 4, 5, 6\}; \bar{C} = \{3\};$
- 10) $\min(U_{13} = 5; U_{23} = 6; U_{43} = 5; U_{63} = 10) = U_{13} = U_{43} = 5;$
- 11) $C = \{1, 2, 3, 4, 5, 6\}; \bar{C} = \{\emptyset\}.$

Получили два варианта минимальной дорожной сети (рис. 3.6) с общей протяженностью $(1 + 3 + 4 + 3 + 5) = 16$.

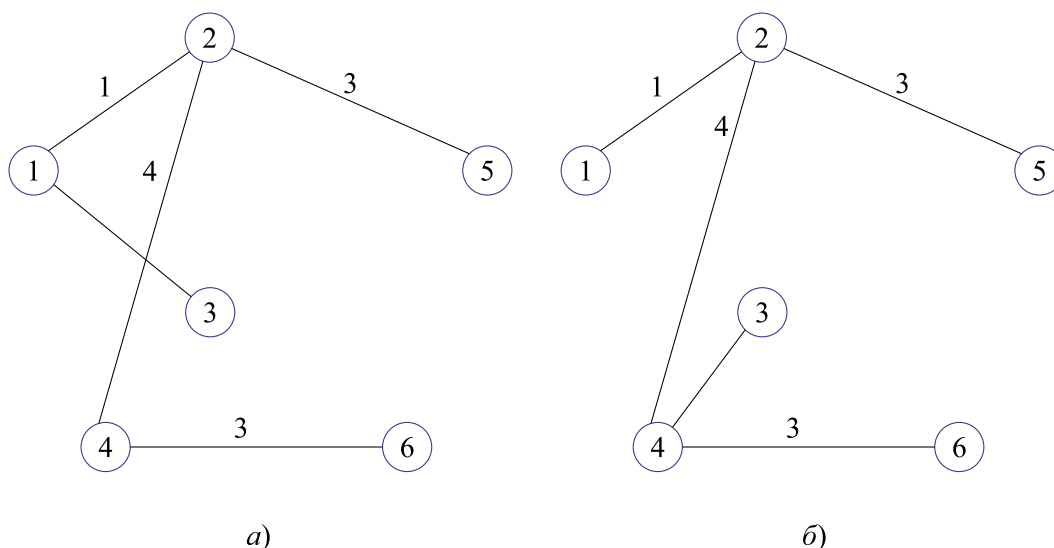


Рис. 3.6 – Варианты минимальной сети:
а) с дугой (1, 3); б) с дугой (3, 4)

Следует отметить, что при любом выборе начальной вершины получается одно и то же оптимальное решение.

.....

3.5 Задача о кратчайшем пути

Задача о кратчайшем пути состоит в нахождении связанных между собой дуг на транспортной сети, которые в совокупности имеют минимальную длину от исходного пункта до пункта назначения. Разработаны алгоритмы нахождения кратчайшего пути — в сетях без циклов и с циклами.

Алгоритм для сетей без циклов

В данном алгоритме представлена идея рекурсивных вычислений, на основе которой в п. 6.2 будут даны алгоритмы решения задач динамического программирования. Рассмотрим следующий пример [1].

**Пример 3.5**

Рассмотрим ориентированную сеть без циклов, изображенную на рисунке 3.7. Вершины сети пронумерованы согласно послойному разбиению графа (см. п. 7.2) таким образом, что вершины с большим номером не могут стоять позади вершин с меньшим номером, внутри слоя нумерация вершин может быть произвольной. Цифры над стрелками характеризуют расстояние на сети между смежными вершинами. Последовательность этапов (шагов) расчета расстояний между начальной вершиной и остальными вершинами определяется послойной разбивкой графа.

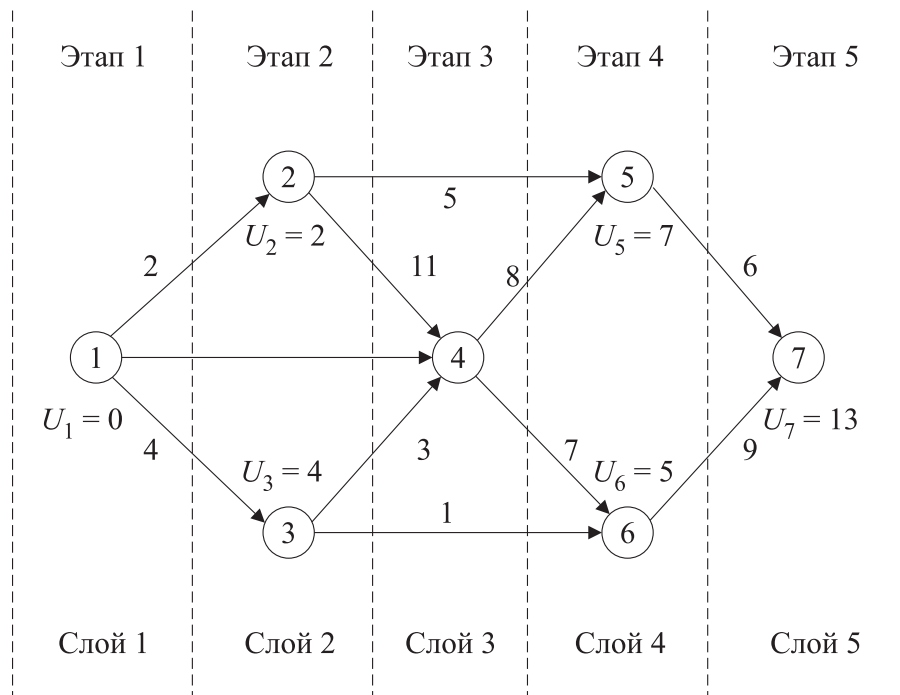


Рис. 3.7 – Ориентированная сеть

Введем обозначения:

- d_{ij} — расстояние на сети по дуге (i, j) ;
- U_j — кратчайшее расстояние от вершины 1 до вершины j , $U_1 = 0$.

Расстояние u_j от вершины 1 до вершины j определяется по формуле:

$$U_j = \min_i \{U_i + d_{ij}\},$$

т. е. кратчайшее расстояние до вершины j определяется после оценки кратчайшего расстояния непосредственно до предыдущей вершины U_i . В этом и заключается рекурсивность вычислений. Вычислим оценки U_i и найдем кратчайший путь.

Этап 1. $U_1 = 0$.

Этап 2. $U_2 = U_1 + d_{12} = 0 + 2 = 2$; $U_3 = U_1 + d_{13} = 0 + 4 = 4$.

Этап 3. $U_4 = \min\{U_1 + d_{14}, U_2 + d_{24}, U_3 + d_{34}\} = \min\{0 + 10, 2 + 11, 4 + 3\} = 7$.

Этап 4. $U_5 = \min\{U_2 + d_{25}, U_4 + d_{45}\} = \min\{2 + 5, 7 + 8\} = 7$.

Этап 5. $U_7 = \min\{U_5 + d_{57}, U_6 + d_{67}\} = \min\{7 + 6, 5 + 9\} = 13$.

Минимальное расстояние между вершинами 1 и 7 равно 13, а соответствующий маршрут определяется дугами (1, 2), (2, 5), (5, 7). Маршрут находится обратным ходом от вершины 7 послойно по дугам, для которых выполняется условие $U_j - d_{ij} = U_i$.

Алгоритм для сетей с циклами

Алгоритм нахождения кратчайшего пути на сети, содержащей циклы, также основан на рекурсивных вычислениях. Однако в силу того, что сеть является неориентированной, разбить ее на слои невозможно и нет четкого перехода от вершины к вершине, то и алгоритмы несколько сложнее предыдущего. Рассмотрим следующий пример.



Пример 3.6

Рассмотрим сеть на рисунке 3.8. Первые цифры в скобках над дугами означают расстояние d_{ij} между вершинами i и j ($i < j$), вторая цифра — обратное расстояние d_{ji} .

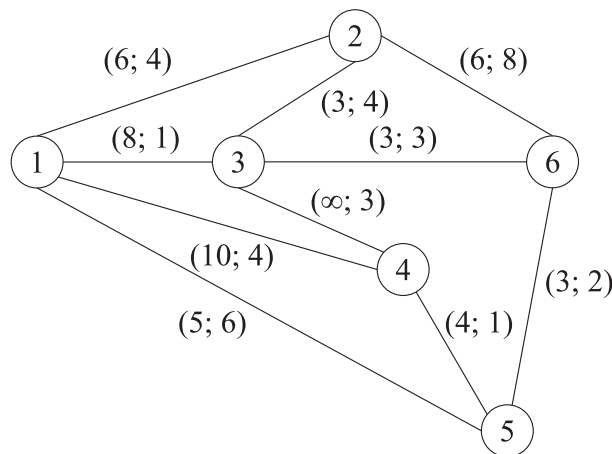


Рис. 3.8 – Транспортная сеть

Алгоритм определения кратчайшего пути на сети с циклами до вершины j заключается вначале в рекурсивном итеративном вычислении кратчайших расстояний от исходной вершины «1» через различные соседние вершины i по формуле:

$$v_j = \min_{i < j} \{U_i + d_{ij}\},$$

где v_j — кратчайшее расстояние до вершины j ; u_i — кратчайшее расстояние до вершины i ; d_{ij} — длина дуги (i, j) .

Таблица 3.14

i	j						u_i
	1	2	3	4	5	6	
1	—	$\begin{smallmatrix} 0 \\ 6 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 8 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 10 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 5 \end{smallmatrix}$		0
2	$\begin{smallmatrix} -10 \\ 4 \end{smallmatrix}$	—	$\begin{smallmatrix} -1 \\ 3 \end{smallmatrix}$			$\begin{smallmatrix} -4 \\ 6 \end{smallmatrix}$	6
3	$\begin{smallmatrix} -9 \\ 1 \end{smallmatrix}$	$\begin{smallmatrix} -6 \\ 4 \end{smallmatrix}$	—	∞		$\begin{smallmatrix} -3 \\ 3 \end{smallmatrix}$	8
4	$\begin{smallmatrix} -14 \\ 4 \end{smallmatrix}$		$\begin{smallmatrix} -8 \\ 6 \end{smallmatrix}$	—	$\begin{smallmatrix} -9 \\ 4 \end{smallmatrix}$	$\begin{smallmatrix} -4 \\ 2 \end{smallmatrix}$	10 6
5	$\begin{smallmatrix} -11 \\ 6 \end{smallmatrix}$			$\begin{smallmatrix} +4 \\ 1 \end{smallmatrix}$	—	$\begin{smallmatrix} 0 \\ 3 \end{smallmatrix}$	5
6		$\begin{smallmatrix} -10 \\ 8 \end{smallmatrix}$	$\begin{smallmatrix} -3 \\ 3 \end{smallmatrix}$	$\begin{smallmatrix} -7 \\ 5 \end{smallmatrix}$	$\begin{smallmatrix} -5 \\ 2 \end{smallmatrix}$	—	8
v_j	0	6	8	10 6	5	8	

$$v_1 = u_1 = 0.$$

$$v_2 = 0 + 6 = 6; u_2 = 6.$$

$$v_3 = \min(0 + 8; 6 + 3) = 8; u_3 = 8.$$

$$v_4 = \min(0 + 10; 8 + \infty) = 10; u_4 = 10.$$

$$v_5 = \min(0 + 5; 10 + 4) = 5; u_5 = 5.$$

$$v_6 = \min(6 + 6; 8 + 3; 10 + 2; 5 + 3) = 8; u_6 = 8.$$

Проверим через найденные оценки выполнение признака оптимальности решения задачи. Для этого, двигаясь построчно, определим для каждого маршрута (i, j) величину $\Delta_{ij} = v_j - u_i - d_{ij}$, которая должна быть в случае оптимальности решения не положительной, ($\Delta_{ij} \leq 0$). Занесем их также в таблицу в правый угол соответствующей клетки.

Условие оптимальности нарушается для $i = 5$ при $j = 4$. Величина v_4 корректируется $v_4 = u_5 + d_{54} = 5 + 1 = 6$; $u_4 = v_4 = 6$. Далее в последующих вычислениях, т. е. для $i = 6$, используются измененные значения. Других клеток с положительными значениями Δ_{ij} нет.

Исходные данные d_{ij} и новые оценки v_j и u_i показаны в таблице 3.15. Вычислим по ним величины Δ_{ij} и занесем в таблицу. Как видно, среди них нет положительных.

Кратчайшее расстояние от вершины 1 до вершины 6 равно 8 единицам. Определяем этот маршрут, начиная с шестого столбца и двигаясь к первой строке через элементы, у которых $\Delta_{ij} = 0$. Таковыми маршрутами будут:

$$(1, 5), (5, 6) \text{ и } (1, 5), (5, 4), (4, 6).$$

Таблица 3.15

i	j						u_i
	1	2	3	4	5	6	
1	—	$\begin{smallmatrix} 0 \\ 6 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 8 \end{smallmatrix}$	$\begin{smallmatrix} -4 \\ 10 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 5 \end{smallmatrix}$		0
2	$\begin{smallmatrix} -10 \\ 4 \end{smallmatrix}$	—	$\begin{smallmatrix} -1 \\ 3 \end{smallmatrix}$			$\begin{smallmatrix} -4 \\ 6 \end{smallmatrix}$	6
3	$\begin{smallmatrix} -9 \\ 1 \end{smallmatrix}$	$\begin{smallmatrix} -6 \\ 4 \end{smallmatrix}$	—	∞		$\begin{smallmatrix} -3 \\ 3 \end{smallmatrix}$	8
4	$\begin{smallmatrix} -10 \\ 4 \end{smallmatrix}$		$\begin{smallmatrix} -4 \\ 6 \end{smallmatrix}$	—	$\begin{smallmatrix} -9 \\ 4 \end{smallmatrix}$	$\begin{smallmatrix} 0 \\ 2 \end{smallmatrix}$	6
5	$\begin{smallmatrix} -11 \\ 6 \end{smallmatrix}$			$\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}$	—	$\begin{smallmatrix} 0 \\ 3 \end{smallmatrix}$	5
6		$\begin{smallmatrix} -10 \\ 8 \end{smallmatrix}$	$\begin{smallmatrix} -3 \\ 3 \end{smallmatrix}$	$\begin{smallmatrix} -7 \\ 5 \end{smallmatrix}$	$\begin{smallmatrix} -5 \\ 2 \end{smallmatrix}$	—	8
v_j	0	6	8	6	5	8	

3.6 Задача о минимальном потоке

Рассматривается задача определения максимального потока между двумя выделенными вершинами связной сети. Пусть задана сеть $G(I, U)$ с одним истоком Z и одним стоком S , на которой задана функция R пропускных способностей дуг r_{ij} .

Необходимо найти поток по дугам x_{ij} , максимизирующий мощность потока V :

$$\max: V = \sum_k x_{zk} = \sum_k x_{ks},$$

при ограничениях:

- 1) поток по дуге в противоположных направлениях отличается по знаку

$$x_{ij} = -x_{ji};$$

- 2) поток по любой дуге не может быть больше ее пропускной способности

$$0 \leq x_{ij} \leq r_{ij};$$

- 3) для любой вершины, не являющейся ни источником, ни стоком, величина суммарного входного потока должна быть равна величине суммарного выходного потока $\sum_i x_{ik} - \sum_j x_{kj} = 0$, для всех $k \neq z, s$;
- 4) мощность исходящего потока для источника z должна совпадать с мощностью входящего потока для стока s $\sum_k x_{zk} = \sum_k x_{ks}$.

Поток x_{ij} , $i, j = \overline{1, n}$ будем задавать матрицей $X = \|x_{ij}\|$. Очевидно, на главной диагонали этой матрицы стоят нули $x_{ii} = 0$, а ниже главной диагонали — числа, имеющие противоположные знаки по отношению к числам, расположенным в матрице симметрично относительно главной диагонали.

Алгоритм решения задачи

Прежде дадим определения. Ребра, для которых выполняются неравенства $x_{ij} < r_{ij}$, называются ненасыщенными, а ребра, для которых выполняются равенства $x_{ij} = r_{ij}$, — насыщенными.

Для определения минимального потока по дугам x_{ij} из источника z в сток s используются следующие шаги.

Шаг 1. Построить начальный допустимый поток $X(K)$, $K = 0$ (K — номер итерации). Решение можно начать с нулевого потока. Однако с целью ускорения решения следует начать с возможно большего начального потока.

Шаг 2. Определяем ненасыщенные дуги по матрице

$$N(K) = \|n_{ij}\| = \|r_{ij} - x_{ij}(K)\|.$$

Шаг 3. Выделяем путь L_N , состоящий из ненасыщенных дуг, ведущий из Z в 1. Если его нет, то $X(K)$ — оптимальное решение и на шаг 5.

Шаг 4. Увеличиваем поток $X(K)$ по этому пути L_N на величину $\theta = \sum_{i,j \in L_N} \{n_{ij}\}$, $K = K + 1$, получаем новый поток $X(K)$ и на шаг 2.

Шаг 5. Определяем максимальный поток в сети из z в s .

$$V = \sum_k x_{ks} = \sum_k x_{zk}.$$

**Пример 3.7**

Рассмотрим сеть на рисунке 3.10 с данными пропускными способностями.

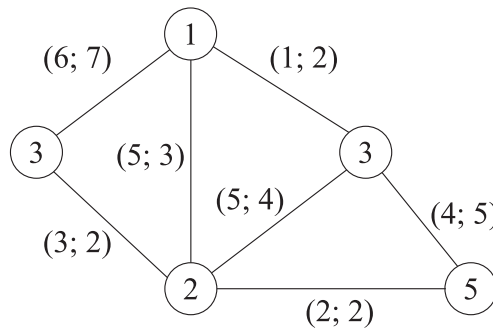


Рис. 3.10 – Транспортная сеть

Соответствующая матрица R пропускных способностей приведена в таблице 3.16. Построим начальный допустимый план X^0 .

По пути $L_1: z \rightarrow 1 \rightarrow 3 \rightarrow s$ можно пустить поток:

$$\theta_1 = \min_{i,j \in L_1} \{r_{ij}\} = \min\{6; 1; 4\} = 1,$$

по пути $L_2: z \rightarrow 2 \rightarrow s$ можно пустить поток:

$$\theta_1 = \min_{L_2} \{3; 2\} = 2.$$

Таблица 3.16 – Матрица пропускных способностей R

$$R = \begin{array}{c|ccccc} & z & 1 & 2 & 3 & s \\ \hline z & 0 & 6 & 3 & & \\ 1 & 7 & 0 & 5 & 1 & \\ 2 & 2 & 3 & 0 & 5 & 2 \\ 3 & & 2 & 4 & 0 & 4 \\ s & & & 2 & 5 & 0 \end{array}$$

Тогда матрица X^0 будет равна

$$X^0 = \begin{array}{c|ccccc} & z & 1 & 2 & 3 & s \\ \hline z & 0 & 1 & 2 & 0 & 0 \\ 1 & -1 & 0 & & 1 & \\ 2 & -2 & & 0 & & 2 \\ 3 & & -1 & & 0 & 1 \\ s & & & -2 & -1 & 0 \end{array} .$$

Матрица N^0 будет равна

$$N^0 = R - X^0 = \begin{array}{c|ccccc} & z & 1 & 2 & 3 & s \\ \hline z & 0 & 5 & 1 & & \\ 1 & 8 & 0 & 5 & 0 & \\ 2 & 4 & 3 & 0 & 5 & 0 \\ 3 & & 3 & 4 & 0 & 3 \\ s & & & 4 & 6 & 0 \end{array} .$$

Находим путь по ненасыщенным дугам $z \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow s$.

Величина $\theta = \min\{5; 5; 5; 3\} = 3$. Увеличиваем поток по найденному пути, получаем X' , затем N' .

В матрице N^1 нет пути от z до s по ненасыщенным дугам. Тогда матрица X^1 является матрицей максимальных потоков дуг. Максимальный поток в сети будет равен $x_{z1} + x_{z2} = x_{21} + x_{31} = 6$. Его величина может быть определена через величину пропускной способности наименьшего разреза сети [14].

$$X^1 = \begin{array}{c|ccccc} & z & 1 & 2 & 3 & s \\ \hline z & 0 & 4 & 2 & & \\ 1 & -4 & 0 & 3 & 1 & \\ 2 & -2 & -3 & 0 & 3 & 2 \\ 3 & & -1 & -3 & 0 & 4 \\ s & & & -2 & -4 & 0 \end{array} . \quad N^1 = \begin{array}{c|ccccc} & z & 1 & 2 & 3 & s \\ \hline z & 0 & 2 & 1 & & \\ 1 & 11 & 0 & 2 & 0 & \\ 2 & 4 & 6 & 0 & 2 & 0 \\ 3 & & 3 & 7 & 0 & 0 \\ s & & & 4 & 9 & 0 \end{array} .$$

Разрезом сети называется множество дуг, соединяющих два подмножества вершин сети Z и S , в одно из которых входит поток $z \in Z$, а в другое — сток $s \in S$. При этом объединение $Z \cup S = I$, а пересечение $Z \cap S = \emptyset$. Пропускная способность разреза $r(Z, S)$ равна сумме пропускных способностей дуг, попавших в разрез: $r(Z, S) = \sum_{i \in Z} \sum_{j \in S} r_{ij}$. Поскольку для любой дуги $x_{ij} \leq r_{ij}$, то поток по разрезу не превосходит пропускной способности этого разреза, т. е. $\sum_{i \in Z} \sum_{j \in S} x_{ij} \leq \sum_{i \in Z} \sum_{j \in S} r_{ij}$.

Следовательно, разрез с минимальной пропускной способностью будет ограничивать пропускную способность сети. Поэтому максимальный поток равен пропускной способности наименьшего разреза сети $V = \min r(Z, S)$.

.....



Контрольные вопросы по главе 3

.....

1. Дайте содержательную и математическую постановку транспортной задачи линейного программирования (ТЗЛП).
2. Можно ли решить ТЗЛП симплекс-методом?
3. Что означают фиктивные переменные в матрице решений несбалансированной ТЗЛП?
4. Какие способы определения начального опорного плана ТЗЛП Вы знаете?
5. Сколько базисных переменных находится в решении ТЗЛП?
6. В чем сущность распределительного метода решения ТЗЛП?
7. На какой теореме основан алгоритм метода потенциалов решения ТЗЛП?
8. В чем отличие метода потенциалов от распределительного метода?
9. Укажите способы решения ТЗЛП с промежуточными пунктами.
10. Можно ли решить задачу о назначениях методом, используемым для решения ТЗЛП?

Глава 4

ДИСКРЕТНЫЕ ЗАДАЧИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

4.1 Классификация моделей и методов дискретного программирования. Примеры задач

Общая задача линейного дискретного целочисленного программирования имеет вид:

$$\max(\min): Z = \sum_{j=1}^n c_j x_j, \quad (4.1)$$

$$\sum_{j=1}^n a_{ij} x_j R b_i, \quad i = \overline{1, m}, \quad (4.2)$$

$$x_j \geq 0, \quad j = \overline{1, n}, \quad (4.3)$$

$$x_j — \text{целые}, \quad j = \overline{1, n}. \quad (4.4)$$

Задача (4.1)–(4.4) называется **полностью** целочисленной задачей линейного программирования, т. к. на все переменные положено условие целочисленности (ограничение (4.4)). Когда это условие относится лишь к некоторым переменным, задача называется **частично** целочисленной.

По структуре математической модели дискретные задачи линейного программирования (ДЗЛП) разделяют на классы [8]:

1. Транспортные задачи (если исходные данные целочисленные) и задачи транспортного типа (о назначениях, с фиксированными доплатами и др.).
2. Задачи с неделимостями (о рюкзаке или ранце, выпуске комплектной продукции и др.).
3. Комбинаторные задачи (развития и размещения производства и складской системы, задача с дихотомией, задача о коммивояжере и др.).

В свою очередь каждый тип моделей подразделяется на статические или динамические, детерминированные или стохастические. Наиболее изучен класс задач детерминированного типа.

В силу того что требование целочисленности (4.4) нельзя выразить линейными неравенствами, в целом задача (4.1)–(4.4) не является классической задачей линейного программирования и в чистом виде алгоритмы ЗЛП для решения ДЗЛП не находят своего применения.

В отличие от задач линейного программирования, решение ДЗЛП в ряде случаев весьма затруднительно. Во всех случаях целочисленная задача, казалось бы, может быть решена обычными методами с отброшенным условием целочисленности и с последующим округлением нецелых переменных в ответе. Однако такое округление может привести к результату, далекому от оптимального целочисленного решения (рис. 4.1). Подобный подход допустим с практической точки зрения в тех случаях, когда значения переменных, образующих оптимальное решение, достаточно велики и погрешностями округления можно пренебречь.

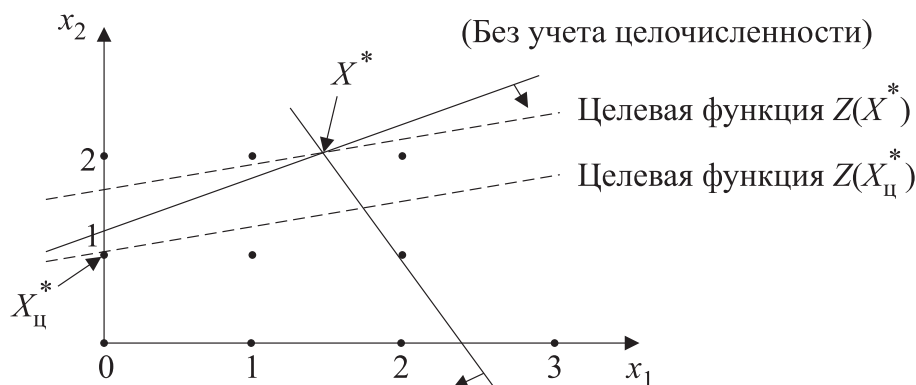


Рис. 4.1 – Геометрическая интерпретация ЗЛП

Выделяют классические задачи целочисленной оптимизации с ярко выраженными целочисленными переменными, такие как задача о назначениях, рюкзаке, коммивояжере, и линейные целочисленные задачи математического программирования (4.1)–(4.4). А также задачи математического программирования, в постановку которых введены дополнительные переменные булевого типа на те или иные условия операций, такие как транспортная задача с фиксированными доплатами, задачи с дихотомией, планирования производственной линии и др. [8].

Приведем некоторые примеры задач.

Задача о рюкзаке

Имеется m ($i = \overline{1, m}$) отсеков вместимостью b_i , оборудованных для перевозки n ($j = \overline{1, n}$) видов грузов. Количество каждого вида груза не ограничено. Каждый вид груза j может быть помещен в любом из отсеков i , но со своим расходным коэффициентом a_{ij} . Полезность j -го вида груза равна C_j . Требуется найти план x_j набора грузов для их перевозки максимальной полезности.

Модель задачи будет следующей:

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

при ограничениях:

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

$$x_j \geq 0, \quad j = \overline{1, n}.$$

Частным случаем задачи о рюкзаке является случай, когда для перевозки имеется один отсек объемом b , а каждый вид груза представлен одним экземпляром.

Модель задачи примет вид:

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

при ограничениях:

$$\sum_{j=1}^n a_j x_j \leq b,$$

$$x_j = \begin{cases} 1, & \text{если груз } j \text{ помещается в рюкзак;} \\ 0, & \text{в противном случае.} \end{cases}$$

Для решения задачи о рюкзаке можно воспользоваться приближенным методом, основанным на правиле «мал золотник, да дорог». Для этого упорядочивают грузы с отношением c_j/a_j по убыванию. Просматривают их слева направо, помещая в рюкзак, пока есть место. Если очередной груз с объемом a_j не вмещается в рюкзак, то откладывают его в сторону и переходят к следующему, пока не пересмотрят все либо не заполнится рюкзак. Такой подход в ряде случаев дает хороший результат. Более точные методы будут рассмотрены ниже.

Транспортная задача с фиксированными доплатами

Имеется m пунктов производства с объемами $a_i (i = \overline{1, m})$ и n пунктов потребления с объемами $b_j (j = \overline{1, n})$. Требуется определить объемы поставок x_{ij} , удовлетворяющие ограничениям транспортной задачи

$$\sum_{j=1}^n x_{ij} = a_i, \quad i = \overline{1, m}, \quad \sum_{i=1}^m x_{ij} = b_j, \quad j = \overline{1, n},$$

и минимизирующие целевую функцию

$$Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij}(x_{ij}),$$

где

$$c_{ij}(x_{ij}) = \begin{cases} d_{ij} + c_{ij}x_{ij}, & \text{если } x_{ij} > 0; \\ 0, & \text{если } x_{ij} = 0, \end{cases}$$

c_{ij} — стоимость перевозки единицы груза от i -го поставщика j -му потребителю; d_{ij} — доплаты за аренду транспортных средств, на строительство подъездных путей и т. д.

Данную задачу можно свести к задаче частично целочисленного программирования. Вводим дополнительные переменные:

$$y_{ij} = \begin{cases} 0, & \text{если } x_{ij} = 0; \\ 1, & \text{если } x_{ij} > 0. \end{cases}$$

Увяжем переменные x_{ij} и y_{ij} отношением $x_{ij} \leq M_{ij} y_{ij}$, которое выполняется всегда при $M_{ij} = \min(a_i, b_j)$.

Тогда исходную задачу можно сформулировать следующим образом:

$$\min : Z = \sum_{i=1}^m \sum_{j=1}^n (d_{ij}y_{ij} + c_{ij}x_{ij}),$$

при ограничениях:

$$\begin{aligned} \sum_{j=1}^n x_{ij} &= a_i, \quad i = \overline{1, m}; \\ \sum_{i=1}^m x_{ij} &= b_j, \quad j = \overline{1, n}; \\ x_{ij} &\leq M_{ij}y_{ij}, \quad i = \overline{1, m}, \quad j = \overline{1, n}; \\ x_{ij} &\geq 0, \quad i = \overline{1, m}, \quad j = \overline{1, n}; \\ y_{ij} &\in \{0, 1\}, \quad i = \overline{1, m}, \quad j = \overline{1, n}. \end{aligned}$$

Сформулированную частично целочисленную задачу с булевыми переменными можно решить приближенным способом путем сведения переменной y_{ij} к непрерывной переменной $0 \leq y_{ij} \leq 1$ через отношение $y_{ij} = \frac{x_{ij}}{M_{ij}}$ и перехода к классической транспортной задаче с целевой функцией

$$Z = \sum_{i=1}^m \sum_{j=1}^n \left(\frac{d_{ij}}{M_{ij}} + c_{ij} \right) x_{ij}.$$

Методы целочисленной оптимизации можно условно разбить на методы:

- 1) отсечения;
- 2) комбинаторные;
- 3) приближенные.

В основе методов отсечения лежит идея сужения области допустимых решений задачи (4.1)–(4.3) с ослабленными ограничениями, т. е. без учета условия (4.4) целочисленности, путем введения специальных дополнительных ограничений (отсекающих плоскостей) до тех пор, пока полученное решение не станет целочисленным.

Дополнительное отсекающее ограничение обладает тем свойством, что оно отсекает найденное оптимальное нецелочисленное решение (4.1)–(4.3) и не затрагивает ни одной из целочисленных точек задачи (4.1)–(4.4).

Метод отсекающих плоскостей, разработанный Р. Гомори, включает два алгоритма: один — для решения полностью целочисленной задачи, а второй — частично целочисленной задачи. В подразделе 4.2 будет дан алгоритм Гомори для решения полностью целочисленной задачи. Другой алгоритм можно найти в литературе [17].

В основе комбинаторных методов лежит идея организованного перебора всех допустимых целочисленных решений. Наиболее известным комбинаторным методом является метод ветвей и границ, предложенный в работе А. Лэнда и А. Дойга в 1960 году применительно к задаче целочисленного линейного программирования. Второе рождение метода связано с работой Литтла и других авторов (Мерти, Суини, Херелом, 1963 г.), посвященной задаче коммивояжера.

4.2 Метод отсечения. Алгоритм Гомори

Пусть дана задача полностью целочисленного линейного программирования (4.1)–(4.4). Алгоритм метода отсечений состоит из следующих этапов:

- 1) решается ЗЛП (4.1)–(4.3) с отброшенным условием целочисленности (4.4), если она неразрешима, то задача (4.1)–(4.4) тоже решения не имеет;
- 2) если условие целочисленности выполняется по всем переменным, то найденное решение есть решение задачи (4.1)–(4.4), иначе
- 3) строится дополнительное отсекающее ограничение, включается в систему ограничений (4.2)–(4.3), и на этап 1.

Для решения полностью целочисленной задачи ЛП Гомори предложено каждый раз на этапе 3 делать дополнительное ограничение для нецелой переменной с наибольшей дробной частью.

Предположим, что задача с отброшенным условием целочисленности решена. Рассмотрим i -ю строку оптимальной симплексной таблицы, которой соответствует нецелое решение β_i базисной переменной x_i . Пусть R — множество индексов j , которые соответствуют небазисным переменным. Тогда переменная x_i может быть выражена через небазисные переменные $x_j, j \in R$:

$$x_i = \beta_i - \sum_{j \in R} d_{ij} x_j, \quad \beta_i — \text{нецелое.} \quad (4.5)$$

Обозначим наибольшую целую часть числа a , его не превосходящую, через $[a]$, ($a \geq [a]$), а дробную положительную часть — через $\{a\}$. Очевидно, $a = [a] + \{a\}$. Например, если $a = 4,7$, то $[a] = 4$; $\{a\} = 0,7$; если $a = -8,6$, то $[a] = -9$, $\{a\} = 0,4$.

Выразим базисную переменную x_i (4.5) в виде суммы целой и дробной частей.

$$x_i = [\beta_i] + \{\beta_i\} - \sum_{j \in R} ([\alpha_{ij}] + \{\alpha_{ij}\}) x_j \left([\beta_i] - \sum_{j \in R} [\alpha_{ij}] x_j \right) + \left(\{\beta_i\} - \sum_{j \in R} \{\alpha_{ij}\} x_j \right). \quad (4.6)$$

Выражение в левых круглых скобках (4.6) — целое число, и чтобы x_i было целым, необходимо, чтобы выражение в правых круглых скобках (4.6) тоже было целым. Когда выражение $L_i = \{\beta_i\} - \sum_{j \in R} \{\alpha_{ij}\} x_j$ будет целым? Так как $0 \leq \{\beta_i\} < 1$, а $\sum_{j \in R} \{\alpha_{ij}\} x_j \geq 0$, то L_i будет целым числом, если $\{\beta_i\} \leq \sum_{j \in R} \{\alpha_{ij}\} x_j$, т. е.

$$\{\beta_i\} - \sum_{j \in R} \{\alpha_{ij}\} x_j \leq 0. \quad (4.7)$$

Соотношение (4.7) определяет правильное отсечение Гомори.

Задача (4.1)–(4.4) не будет иметь полностью целочисленного решения, если в симплекс-таблице встретится уравнение такое, что β_i — дробное число, а все α_{ij} — целые.



Пример 4.1

Решить задачу ЛП $\max: Z = 3x_1 + x_2$, при ограничениях:

$$3x_1 - 2x_2 \leq 3;$$

$$-5x_1 - 4x_2 \leq -10;$$

$$2x_1 + x_2 \leq 5;$$

$$x, x_2 \geq 0;$$

$$x, x_2 \text{ — целые.}$$

Решение:

Без учета целочисленности оптимальной симплекс-таблицей будет таблица 4.1.

Таблица 4.1 – Конечная симплекс-таблица

Базис	x_1	x_2	x_3	x_4	x_5	Значение
Z	0	0	$\frac{5}{7}$	0	$\frac{3}{7}$	$4\frac{2}{7}$
x_1	1	0	$\frac{1}{7}$	0	$\frac{2}{7}$	$1\frac{6}{7}$
x_2	0	1	$-\frac{2}{7}$	0	$\frac{3}{7}$	$1\frac{2}{7}$
x_4	0	0	$-\frac{3}{7}$	1	$\frac{1}{7}$	$1\frac{3}{7}$

Так как полученное решение не является целочисленным, следует расширить данную таблицу путем введения отсечения Гомори на ту переменную x_i , которой соответствует $\max(\{\beta_1\}, \{\beta_2\})$, т. е. на переменную x_1 . Строке с базисной переменной x_1 соответствует равенство:

$$x_1 + \frac{1}{7} \cdot x_3 + \frac{2}{7} \cdot x_5 = 1\frac{6}{7}.$$

Следовательно, отношение Гомори имеет вид:

$$\frac{6}{7} - \frac{1}{7} \cdot x_3 - \frac{2}{7} \cdot x_5 \leq 0. \quad (4.8)$$

Приведем данное неравенство к каноническому виду и добавим к таблице 4.1 (табл. 4.2):

$$-\frac{1}{7} \cdot x_3 - \frac{2}{7} \cdot x_5 + x_6 = -\frac{6}{7}.$$

Таблица 4.2 – Расширенная симплекс-таблица

Базис	x_1	x_2	x_3	x_4	x_5	x_6	Значение
Z	0	0	$\frac{5}{7}$	0	$\frac{3}{7}$	0	$4\frac{2}{7}$
x_1	1	0	$\frac{1}{7}$	0	$\frac{2}{7}$	0	$1\frac{6}{7}$
x_2	0	1	$-\frac{2}{7}$	0	$\frac{3}{7}$	0	$1\frac{2}{7}$
x_4	0	0	$-\frac{3}{7}$	1	$\frac{1}{7}$	0	$1\frac{3}{7}$
x_6	0	0	$-\frac{1}{7}$	0	$-\frac{2}{7}$	1	$-\frac{6}{7}$

Далее опять будем применять симплекс-алгоритм (двойственный, выводя из базиса x_6) до тех пор, пока базисные значения x_1 и x_2 не станут целыми. Решением задачи будет вектор $X_{\Pi}^* = (1; 2)$.

Покажем на графике (рис. 4.2) отсечение Гомори, выраженное через базисные переменные. Выразим из уравнений канонической формы исходной ЗЛП:

$$3x_1 - 2x_2 + x_3 = 3;$$

$$2x_1 + x_2 + x_5 = 5,$$

переменные x_3 и x_5 и подставим в (4.8). Получим $13 - 7x_1 \geq 6$, т. е. отсечение Гомори, выраженное через базисные переменные, будет $x_1 \leq 1$.

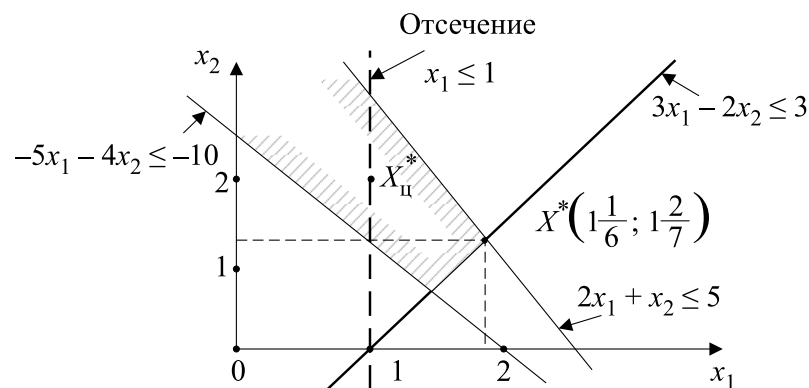


Рис. 4.2 – Геометрическая интерпретация отсечения

4.3 Метод ветвей и границ

4.3.1 Общая задача целочисленного программирования

Несмотря на то, что методы отсечений являются надежным средством решения целочисленных задач, они не подходят для задач большой размерности в силу большого объема вычислений. Однако идеи методов отсечений оказываются полезными для совершенствования других алгоритмов решения целочисленных задач, в частности методов ветвей и границ [1].

Метод ветвей и границ — один из наиболее эффективных методов решения различных задач комбинаторного типа, алгоритмы решений которых заключаются не в полном, а в частичном переборе вариантов, в организованном поиске оптимума. Суть метода заключается в разбиении множества допустимых решений (в линейном программировании — симплекса) на подмножества (как правило, определяются поэтапно), для каждого из которых определенным образом может быть установлена оценка (граница) достижения экстремума. Поиск решения продолжается каждый раз в том подмножестве (по той ветке), в котором потенциально может лежать лучшее решение.

Согласно общей идее метода сначала решается ЗЛП без учета усложненности. Пусть x_k — целочисленная переменная, значение которой x_k^* в оптимальном решении задачи без учета целочисленности является дробным. Интервал $[x_k^*] < x_k < [x_k^*] + 1$ не содержит допустимые целочисленные точки, поэтому его можно исключить из области поиска решения целочисленной задачи. Это делается путем разбиения области допустимых решений нецелочисленной задачи на две области посредством введения дополнительного ограничения на целочисленную переменную для одной задачи $x_k \leq [x_k^*]$ и для другой — $x_k \geq [x_k^*] + 1$.

Введение этих условий (отсекающих плоскостей) в нецелочисленную задачу порождает две не связанные между собой задачи. Исходная задача разветвляется на две подзадачи. Затем каждая подзадача решается как ЗЛП с целевой функцией исходной задачи. Если полученное решение оказывается целочисленным, оно фиксируется через значение целевой функции как наилучшее. Для задачи определена граница, которую надо улучшить. В противном случае подзадача должна быть разбита на две подзадачи путем введения дополнительных ограничений на целочисленность переменной. Как только полученное допустимое целочисленное решение одной из подзадач оказывается лучше имеющегося, оно фиксируется вместо зафиксированного ранее (происходит изменение границы задачи). Процесс ветвления продолжается до тех пор, пока каждая подзадача не приведет к целочисленному решению или пока не будет установлена невозможность улучшения имеющегося решения.



Пример 4.2

Максимизировать $Z = 2x_1 + 3x_2$ при ограничениях [1]:

$$5x_1 + 7x_2 \leq 35;$$

$$4x_1 + 9x_2 \leq 36;$$

$$x_1, x_2 \geq 0 \text{ — целые.}$$

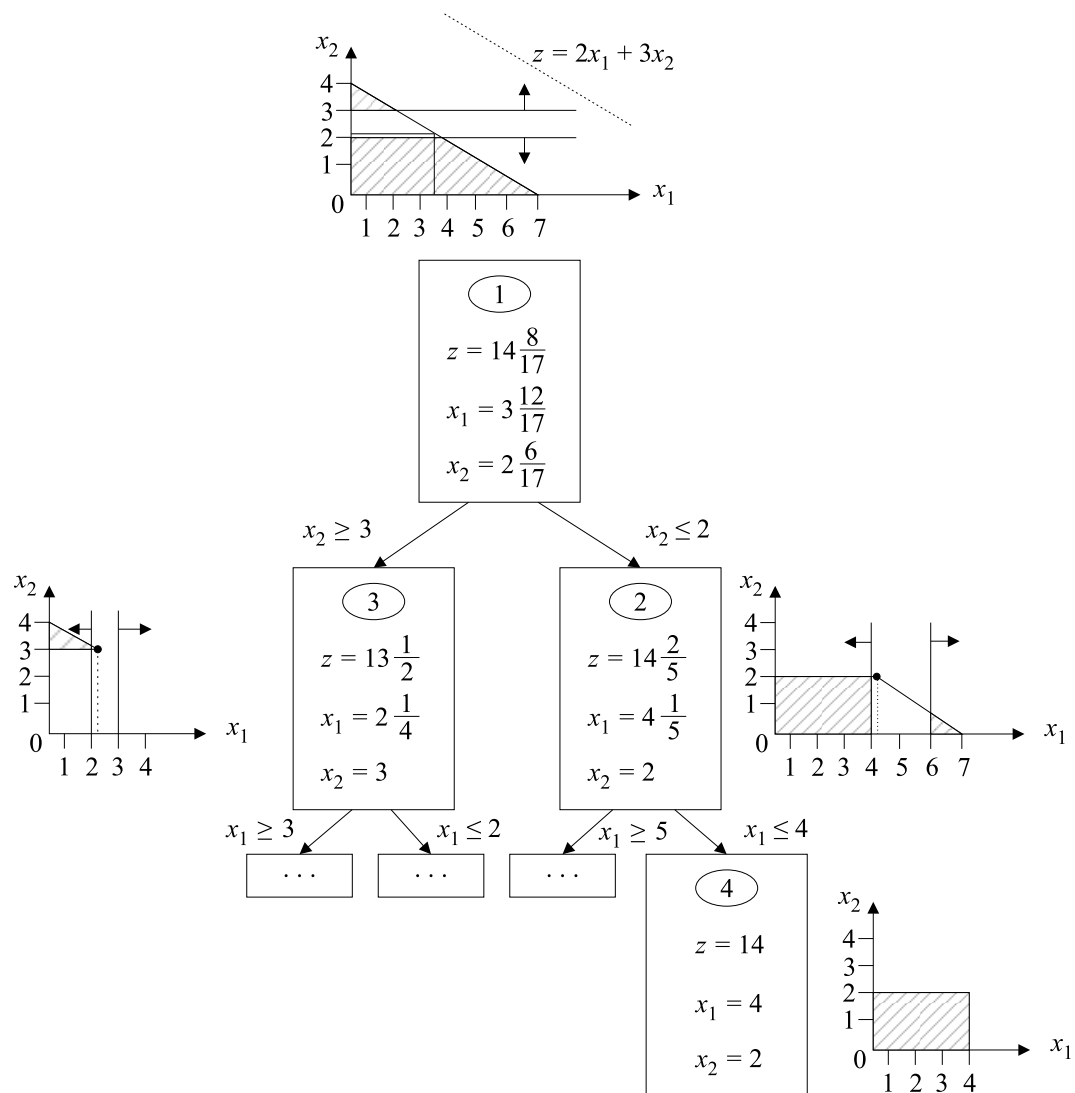


Рис. 4.3 – Фрагмент процесса ветвления поиска решения

Решение:

На рисунке 4.3 показаны начальные этапы процесса разбиения исходной задачи на подзадачи в виде дерева. Исходной является вершина 1, в которой сформулированная задача решается как задача ЛП. Оптимальное решение достигается в точке

с координатами $x_1 = 3\frac{12}{17}$ и $x_2 = 2\frac{6}{17}$. Процесс ветвления далее можно вести по переменной x_1 или x_2 . Выберем ветвление по переменной x_2 . Тогда выбор переменной x_2 порождает две подзадачи, связанные с поиском решения в двух подмножествах, полученных отсекающими ограничениями $x_2 \leq 2$ и $x_2 \geq 3$. На следующем шаге осуществляется выбор одной из подзадач — 2 или 3. К сожалению, вопрос о «наилучшем» способе выбора переменной ветвления или последовательности решения подзадач пока еще не решен, и в каждом конкретном случае решается эмпирически.

Целочисленное решение задачи 4 определяет нижнюю границу $z = 14$ целевой функции. Нет необходимости рассматривать те последующие подзадачи, для которых оптимальные значения z меньше указанной нижней границы. Другим решением задачи может быть точка с координатами $x_1 = 7$, $x_2 = 0$, $z = 14$, если двигаться по ветке с отсекающим ограничением $x_1 \geq 5$.

.....

4.3.2 Задача о коммивояжере

Коммивояжер (посыльный, развозчик заказанной продукции) должен посетить каждый из n пунктов, связанных между собой дорогами, только один раз и вернуться в исходный пункт. Его маршрут должен минимизировать суммарную длину пройденного пути.

Формализуем задачу.

Пусть известна матрица $C = \|c_{ij}\|$ расстояний между пунктами i и j ($i = \overline{1, n}$; $j = \overline{1, n}$; $i \neq j$). В качестве неизвестной величины введем переменную:

$$x_{ij} = \begin{cases} 1, & \text{если коммивояжер из пункта } i \text{ переезжает в пункт } j; \\ 0, & \text{в противном случае.} \end{cases}$$

Модель задачи о коммивояжере будет иметь вид:

$$\min Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}, \quad (4.9)$$

при ограничениях

$$\sum_{i=1}^n x_{ij} = 1, \quad j = \overline{1, n}; \quad (4.10)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = \overline{1, n}; \quad (4.11)$$

$$x_{ij} \in \{0, 1\}, \quad i = \overline{1, n}; \quad j = \overline{1, n}. \quad (4.12)$$

Еще одно ограничение сформулируем следующим образом: искомые переменные $x_{ij} \in \{0, 1\}$ должны образовывать полный контур, включающий все пункты.

Ограничение (4.10) говорит о том, что коммивояжер должен в каждый пункт $j = \overline{1, n}$ заехать только один раз, а ограничение (4.11) — о том, что из каждого пункта $i = \overline{1, n}$ выехать только один раз. Ограничения (4.10)–(4.12) и дополнительное

ограничение на маршруте коммивояжера создают так называемый гамильтоновы контур (по имени ирландского математика У. Гамильтона).

Не в каждом графе существует гамильтоновы контур. Сформулировано достаточное условие существования этого контура: если степень каждой вершины (число ребер, выходящих из вершины) графа, который имеет n вершин ($n \geq 3$), не меньше $n/2$, то на этом графе можно построить гамильтоновы контур. Другими словами, если в группе из n человек ($n \geq 3$) каждый имеет по крайней мере $n/2$ знакомых, то всю группу можно усадить вокруг стола таким образом, что каждый из них будет знаком с двумя соседями по столу.

Если сформулированное достаточное условие не выполняется для всех вершин, то это еще не означает, что в графе нет гамильтонова контура. Условие (необходимое) обязательного присутствия данного контура в графе пока не найдено.

Предложено достаточно большое количество методов поиска на графе гамильтоновых контуров минимальной длины, в основе которых лежит подход организованного, отличного от полного, просмотра «перспективных» маршрутов.

Алгоритм исключения подциклов

Если задачу о коммивояжере ограничить только условиями (4.10)–(4.12), то она станет эквивалентной задаче о назначениях. Поэтому есть методы решения задачи, в которых используют алгоритмы решения задачи о назначениях с исключением возможностей образования подциклов [1, 8, 32].



Пример 4.3

Дана следующая матрица расстояний между пятью городами (в матрице M — число, близкое к бесконечности, означает, что переезд из данного пункта в данный пункт закрыт).

1	M	1	25	11	2
2	37	M	26	58	21
3	10	50	M	39	22
4	38	39	24	M	38
5	27	9	32	9	M
	1	2	3	4	5

Решить задачу о коммивояжере как задачу о назначениях.

Решение:

На рисунке 4.4 показана схема ветвления решения задачи о коммивояжере. Решение исходной задачи без условия исключения подциклов как задачи о назначениях дало результаты: $Z = 149$, $x_{31} = x_{14} = x_{43} = x_{52} = x_{25} = 1$. Образовались два подпункта (3–1–4–3) и (5–2–5). В одном из подциклов разрывают дуги, образующие эти циклы. Возьмем подцикл с меньшим числом городов (5–2–5) и разорвем дуги (5–2) и (2–5). Получим столько новых задач, сколько городов в подцикле. Оптимальным решением будет решение подзадачи с меньшим значением целевой функции.

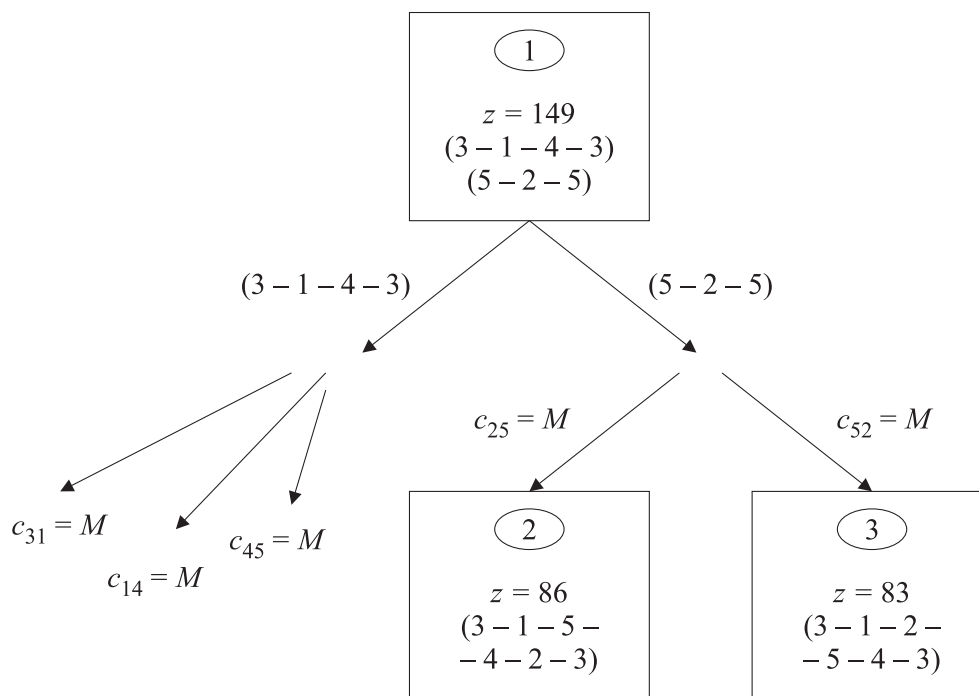


Рис. 4.4 – Схема ветвления решения задачи

Алгоритм ближайшего соседа

Рассмотрим другую группу методов решения задачи. В них круг перспективных маршрутов выстраивается по схеме разветвления возможных направлений движения коммивояжера в виде граф-дерева, а отсечение (отсев) неперспективных маршрутов производится по оценкам (границам) их эффективности.

**Пример 4.4**

Пусть граф, на котором отыскивается кратчайший гамильтонов маршрут, задан матрицей расстояний:

1	M	5	7	3
2	2	M	4	8
3	6	2	M	4
4	3	8	8	M
	1	2	3	4

Алгоритм «ближайшего соседа» по методу ветвей и границ представлен в виде блок-схемы (рис. 4.5). Под оценкой вершины понимается сумма длин дорог, ведущих в данную вершину.

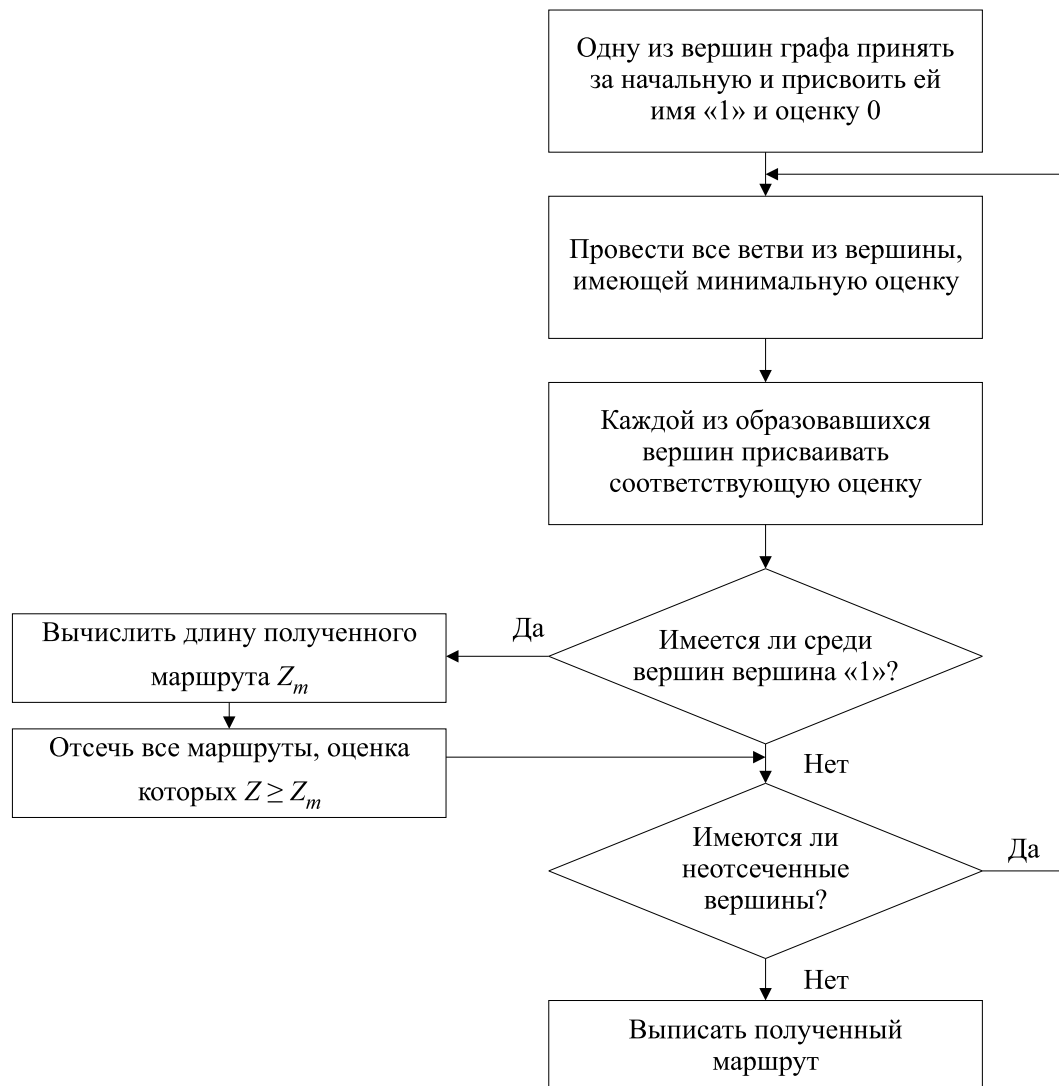


Рис. 4.5 – Блок-схема алгоритма «ближайшего соседа»

Решение:

В ходе решения задачи постепенно, шаг за шагом, будет вырастать дерево, ветви которого отсекаются (рис. 4.6). Оптимальным решением является маршрут (1–4–3–2–1) длиной $Z(x) = c_{14} + c_{43} + c_{32} + c_{21} = 3 + 8 + 2 + 2 = 15$.

Основным недостатком метода является то, что степень приближения к оптимуму зависит от выбора начальной вершины. Поэтому этот алгоритм целесообразно применить, начиная с каждой вершины, и затем выбрать замкнутый контур наименьшей длины.

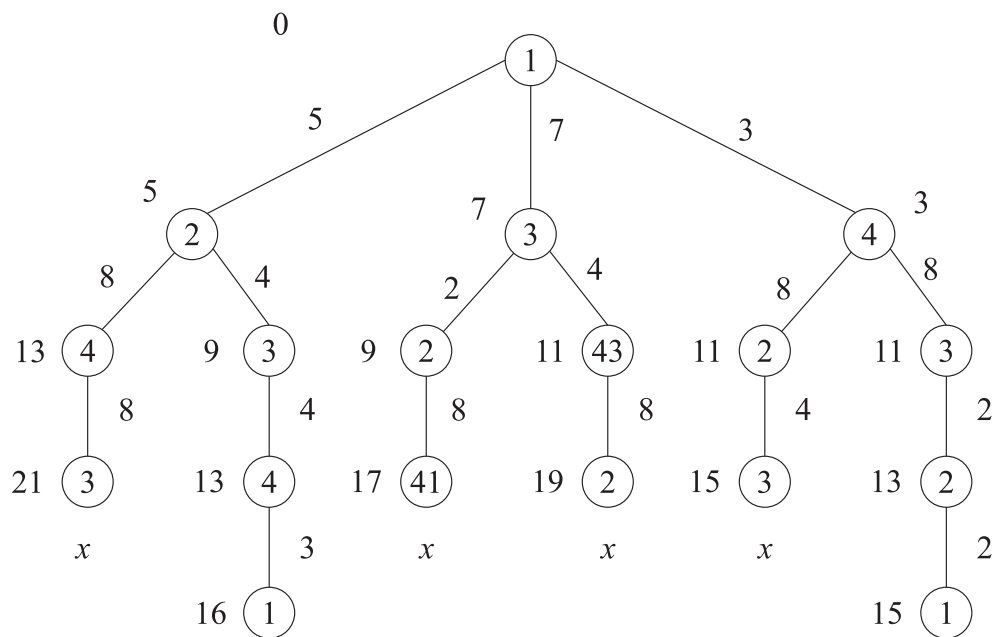


Рис. 4.6 – Дерево решений

Алгоритм Литтла

Для практической реализации метода ветвей и границ применительно к задаче коммивояжера Литтл нашел метод разбиения множества гамильтоновых контуров на подмножества (ветвление) и определения их граничных оценок.

Основная идея метода состоит в том, что сначала строят нижнюю границу длин маршрутов для всего множества гамильтоновых контуров $\varphi(\Omega^0)$. Затем все множество контуров Ω^0 разбивают на два таким образом, чтобы первое подмножество Ω_{ij}^1 состояло из гамильтоновых контуров, содержащих некоторую дугу (ij) , а другое подмножество Ω_{ij}^1 не содержало этой дуги.

Для каждого из подмножеств определяют нижние границы по тому же правилу, что и для первоначального множества гамильтоновых контуров. Полученные оценки $\varphi(\Omega_{ij}^1) \geq \varphi(\Omega^0)$ и $\varphi(\Omega_{ij}^1) \geq \varphi(\Omega^0)$ указывают на первоочередной просмотр соответствующего множества. Далее процесс ветвления продолжается по аналогичному правилу до тех пор, пока не отыщется гамильтонов контур.

Определение нижних границ базируется на следующем утверждении. Если ко всем элементам i -й строки или j -го столбца матрицы добавить или отнять от них число α , то задача останется эквивалентной прежней, т. е. оптимальность маршрута коммивояжера не изменится, а длина любого гамильтонова контура изменится на данную величину α . Это происходит потому, что в маршрут включается только одна дуга по строке i и столбцу j .

Определение нижней границы для множества Ω^0 гамильтоновых контуров $\varphi(\Omega^0)$ производится следующим образом:

1. Находим в каждой строке матрицы $C = \|c_{ij}\|$ минимальный элемент $u_i = \min_j c_{ij}$ и вычитаем его из всех элементов соответствующей строки. Получаем матрицу C' , приведенную по строкам.

2. Находим в каждом столбце матрицы $C' = \|c'_{ij}\|$ минимальный элемент $\delta_j = \min_i c'_{ij}$ и вычитаем его из всех элементов соответствующего столбца. Получаем матрицу C'' , приведенную по строкам и столбцам (в ней обязательно будут нулевые элементы в каждой строке и столбце):

$$C''_{ij} = c'_{ij} - \delta_j.$$

3. Получаем константу приведения γ^0 :

$$\gamma^0 = \varphi(\Omega^0) = \sum_{i=1}^n u_i + \sum_{j=1}^n \delta_j,$$

которая будет нижней границей множества всех допустимых гамильтоновых контуров $\varphi(\Omega^0) \leq Z(x)$.

Если считать, что матрица затрат C оценивается в рублях, то u_i будет означать минимальную сумму рублей, необходимую для того, чтобы выехать из города i . δ_j будет означать минимальную сумму рублей, необходимую для того, чтобы заехать в город j . Тогда $\varphi(\Omega^0) = \sum_i u_i + \sum_j \delta_j$ будет отождествляться с той минимальной суммой рублей, ниже которой проезд по гамильтонову контуру невозможен.

Разбиение множества гамильтоновых контуров на подмножества делается некоторой дугой. Первое подмножество Ω^1_{ij} составляют гамильтоновы контуры, содержащие дугу (i, j) , второе Ω^1_{ij} не содержит дуги (i, j) . Включение дуги (i, j) в контур приводит к сокращению матрицы C на строку i и столбец j , а также к закрытию проезда в обратном направлении, т. е. следует положить $c_{ij} = M$. Сократив размер матрицы C , следует уточнить нижнюю границу $\varphi(\Omega^1_{ij}) \geq \varphi(\Omega^0)$.

Множество Ω^1_{ij} отождествляется с матрицей C , в которой нет гамильтоновых контуров с проездом от i к j , т. е. в матрице C следует положить $c_{ij} = M$. После этого также уточняется нижняя граница $\varphi(\Omega^1_{ij}) \geq \varphi(\Omega^0)$.

Какую же дугу следует включать в гамильтонов контур? Естественно, дугу минимальной длины. В приведенной матрице C'' выясняются все $C'' = 0$ (претенденты для включения в контур). Для каждого элемента $C'' = 0$ определяют степень нулевого элемента θ_{ij} :

$$\theta_{ij} = \min_{j' \neq j} C''_{ij'} + \min_{j' \neq j} C''_{ij'}.$$

О чем говорит степень нулевого элемента (дуги i, j) θ_{ij} ? Обратимся к рисунку 4.7, на котором дан фрагмент некоторого графа. Ему соответствует матрица затрат в рублях (табл. 4.3), где L — некоторое большое число, M — очень большое, $L \ll M$.

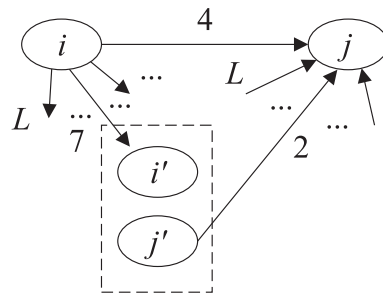


Рис. 4.7 – Фрагмент графа

Таблица 4.3 – Матрица расстояний C

i	M	4	L	7	$u_1 = 4$
j	M	M	M	M	$u_2 = 0$
i'	M	2	M	0	$u_3 = 0$
j'	M	L	0	M	$u_4 = 0$
	i	j	i'	j'	

Сделаем приведение матрицы по строкам и столбцам (табл. 4.4).

Таблица 4.4 – Приведенная матрица

$$C''$$

i	M	0 ⁵	$L - 4$	3
j	M	M	M	M
i'	M	2	M	0
j'	M	L	0	M
	i	j	i'	j'

Степень нулевого элемента (i, j) равна 5, $\theta_{ij} = 5$. Чтобы выехать из города i , минуя маршрут (i, j) , необходимо дополнительно минимум 3 рубля, а чтобы, минуя этот же маршрут (i, j) , заехать в город j , необходимо дополнительно 2 рубля. Таким образом, если мы не поедем по маршруту (i, j) , а поедем в объезд его, то как минимум мы понесем дополнительные затраты, равные 5 рублям. Сам переезд $c_{ij} = 4$, в объезд его затратим $c_{ij} + \theta_{ij} = c_{ij'} + c_{i'j} = 9$ рублей. Следовательно, степень нулевого элемента характеризует дополнительные затраты (штраф за то, что не выберем маршрут (i, j) , а поедем в объезд его), поэтому необходимо выбирать дугу для включения в маршрут с наибольшей степенью нулевого элемента.

После определения нижней границы множества Ω^0 гамильтоновых контуров $\varphi(\Omega^0)$ предпримем следующие шаги.

4. Находим степени нулей для приведенной по строкам и столбцам матрицы C'' , суммируя минимальные элементы строки и столбца, соответствующих этому нулю. Записываем ее в правом углу клетки.

5. Выбираем дугу (i, j) , для которой степень нулевого элемента достигает максимального значения:

$$\theta_{ij} = \max_{c_{kl}} = \theta_{kl}.$$

6. Разбиваем множество Ω^0 на два подмножества Ω_{ij}^1 и $\Omega_{\bar{i}\bar{j}}^1$. Для получения матрицы контуров Ω_{ij}^1 с дугой (i, j) вычеркиваем в матрице C'' строку i и столбец j , а симметричный элемент (i, j) заменяем на бесконечность, $c_{ij} = M$.

7. Приводим матрицу Ω_{ij}^1 по строкам и столбцам. Пусть γ_{ij}^1 — константа приведения. Тогда нижняя граница множества Ω_{ij}^1 определяется так:

$$\varphi(\Omega_{ij}^1) \geq \varphi(\Omega^0) + \gamma_{ij}^1.$$

8. Для получения подмножества гамильтоновых контуров $\Omega_{\bar{i}\bar{j}}^1$ без дуги (i, j) заменяем элемент c_{ij} в матрице C'' на бесконечность $c_{ij} = M$.

9. Делаем приведение матрицы $\Omega_{\bar{i}\bar{j}}^1$. Пусть $\gamma_{\bar{i}\bar{j}}^1$ — константа ее приведения. Тогда нижняя граница множества $\Omega_{\bar{i}\bar{j}}^1$ установится так:

$$\varphi(\Omega_{\bar{i}\bar{j}}^1) = \varphi(\Omega^0) + \gamma_{\bar{i}\bar{j}}^1.$$

10. Сравниваем нижние границы для подмножеств. Если $\varphi(\Omega_{ij}^1) < \varphi(\Omega_{\bar{i}\bar{j}}^1)$, то дальнейшему ветвлению подлежит подмножество Ω_{ij}^1 , и т. д.

11. Ветвление производим до матрицы размерностью 2×2 . Избегая подциклов, определяем гамильтонов контур и его длину.

12. Сравниваем длину гамильтонова контура с нижними границами оборванных ветвей. Если длина контура не превышает их нижних границ, то задача решена. В противном случае развиваем ветвь подмножества с меньшей нижней границей.



Пример 4.5

Обратимся к исходным данным примера 4.4 и найдем гамильтонов контур алгоритма Литтла. Исходное множество всех гамильтоновых контуров задается матрицей C (табл. 4.5).

Таблица 4.5 – Множество Ω^0

1	M	5	7	3	$u_1 = 3$
2	2	M	4	8	$u_2 = 2$
3	6	2	M	4	$u_3 = 2$
4	3	8	8	M	$u_4 = 3$
	1	2	3	4	

$\Rightarrow$

Решение:

1. Делаем приведение матрицы C по строкам и по столбцам (табл. 4.6).

2. Константа приведения $\gamma^0(\Omega^0) = \sum_i u_i + \sum_j v_j = 10 + 2 = 12$. Нижняя граница множества всех гамильтоновых контуров $\varphi(\Omega^0) = \gamma^0(\Omega^0) = 12 \leq z(x)$.

3. Находим степени нулей полностью приведенной матрицы (табл. 4.6).

4. Две дуги имеют максимальные степени нулей $\theta_{14} = \theta_{32} = 4$. Включаем одну из них в маршрут, пусть это будет дуга $(1, 4)$.

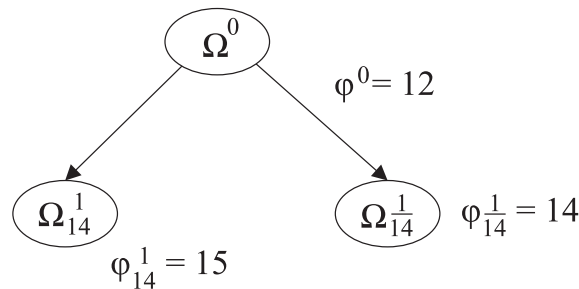
Таблица 4.6 – Приведенное множество $\tilde{\Omega}^0$

$\Rightarrow$

M	2	4	0
0	M	2	6
4	0	M	2
0	5	5	M
0	0	2	0

 $\Rightarrow$

1	M	2	2	0^4
2	0^0	M	0^2	6
3	4	0^4	M	2
4	0^3	5	3	M
	1	2	3	4

Рис. 4.8 – Схема ветвления множества Ω^0

5. Разбиваем множество Ω^0 на два: Ω^1_{14} и Ω^1_{14} . Матрицу множества Ω^1_{14} с дугой (1, 4) получаем из таблицы 4.6 путем вычеркивания строки 1 и столбца 4. Чтобы не допустить образования негамильтонового контура, заменяем элемент (4, 1) на M (табл. 4.7).

Таблица 4.7 – Множество Ω^1_{14}

2	0	M	0	0
3	4	0	M	0
4	$M!$	5	3	3
	1	2	3	

$\Rightarrow$

6. Матрицу гамильтоновых контуров Ω^1_{14} получим из таблицы 4.6 путем замены элемента $C''_{14} = M$ (табл. 4.8).

Таблица 4.8 – Приведенное множество $\tilde{\Omega}^1_{14}$

	2	0^4	M	0
$\Rightarrow$	3	4	0^6	M
	4	M	2	0^2
		1	2	3

7. Делаем приведение матрицы контуров Ω^1_{14} , получаем таблицу 4.9, нижняя граница — $\varphi(\Omega^1_{14}) = \varphi(\Omega^0) + \gamma^1_{14} = 12 + (3 + 0) = 15$.

Таблица 4.9 – Множество Ω_{14}^1

1	M	2	2	$M!$	2
2	0	M	0	6	0
3	4	0	M	2	0
4	0	5	3	M	0
	1	2	3	4	

 $\Rightarrow$

Делаем приведение матрицы контуров Ω_{14}^1 , получаем таблицу 4.10, нижняя граница – $\varphi(\Omega_{14}^1) = 12 + 2 = 14$.

Таблица 4.10 – Приведенное множество $\tilde{\Omega}_{14}^1$
 $\Rightarrow$

1	M	0^0	0^0	M
2	0^0	M	0	6
3	4	0^2	M	2
4	0^3	5	3	M
	1	2	3	4

8. Сравниваем нижние границы, дальнейшему ветвлению подвергаем множество Ω_{14}^1 , т. к. $\varphi(\Omega_{14}^1) > \varphi(\Omega_{14}^1)$.

9. Находим степени нулей матрицы (табл. 4.10). Включаем дугу (4, 1) в маршрут, т. к. $\theta_{41} = 3$.

10. Разбиваем Ω_{14}^1 на Ω_{41}^2 и Ω_{41}^2 :

		2	3	4	u_i			1	2	3	4	
Ω_{41}^2 :	1	0	0	$M!$	0	;	Ω_{41}^2 :	1	M	0	0	M
	2	M	0	6	0			2	0	M	0	6
	3	0	M	2	0			3	4	0	M	2
	v_j	0	0	2				4	M	5	3	M

Множество Ω_{41}^2 далее не рассматривается, т. к. оно не содержит ни одного гамильтонова контура в силу того, что $C_{14}'' = C_{41}'' = M$.

11. Определяем нижнюю границу:

$$\varphi(\Omega_{41}^2) = \varphi(\Omega_{14}^1) + \gamma_{41}^2 = 14 + 2 = 16.$$

12. Возвращаемся на множество гамильтоновых контуров Ω_{14}^1 (табл. 4.8) и оцениваем степени нулей. Включаем дугу (3, 2) в маршрут $\theta_{32} = 6$.

13. Разбиваем Ω_{14}^1 на Ω_{32}^2 и Ω_{32}^2 :

		1	3	u_i				1	2	3	u_i
Ω_{32}^2 :	2	0	$M!$	0	;	Ω_{32}^2 :	2	0	M	0	0
	4	M	0	0			3	4	$M!$	M	4
							4	M	2	0	0

14. Определяем нижние границы:

$$\varphi(\Omega_{32}^2) = \varphi(\Omega_{14}^1) + \gamma_{32}^2 = 15 + 0 = 15.$$

$$\varphi(\Omega_{32}^2) = \varphi(\Omega_{14}^1) + \gamma_{32}^2 = 15 + 6 = 21.$$

Следовательно, ветвлению нужно подвергнуть подмножество Ω_{32}^2 , но его матрица имеет размерность 2×2 . Поэтому в гамильтонов контур следует включить дуги, соответствующие в матрице нулевым элементам, т. е. дуги (2, 1) и (4, 3).

На рисунке 4.9 представлено дерево ветвлений.

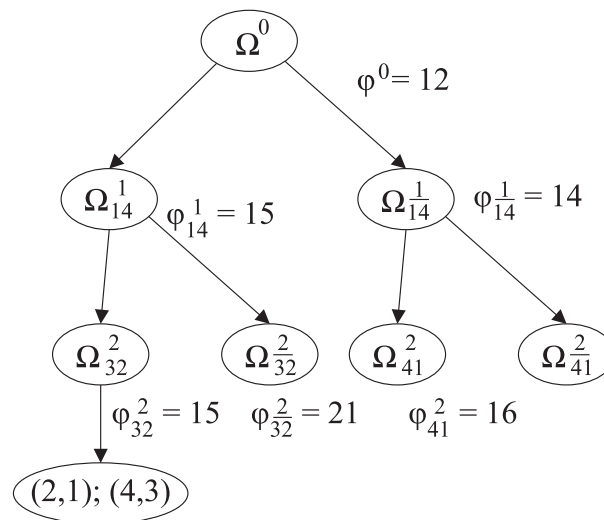


Рис. 4.9 – Дерево ветвлений

Длина гамильтонова контура равна $c_{14} + c_{32} + c_{21} + c_{43} = 3 + 2 + 2 + 8 = 15$. Так как границы оборванных ветвей больше длины контура 1–4–3–2–1, то этот контур имеет наименьшую длину.

.....

В заключение раздела остановимся еще на следующих задачах, для решения которых также используются алгоритмы Литтла.

Решение задачи коммивояжера на максимум

В этом случае ее легко сводят к задаче коммивояжера на минимум. Для этого поступают так. Переходят от матрицы эффективности $C = \|c_{ij}\|$ к матрице затрат $\tilde{C} = \|\tilde{c}_{ij}\|$, где $\tilde{c}_{ij} = \max_i c_{ij} - c_{ij}$. Затем решают задачу на минимум с матрицей $\tilde{C}$. Как было показано (задача о назначениях), решение при этом переходе не изменится.

Решение задачи коммивояжера с заданным началом пути

В ряде случаев начальная вершина графа задана, а коммивояжеру нужно не объехать по контуру все города, а отправиться из заданного города, не возвращаясь обратно. Казалось бы, гамильтонов путь (не контур) минимальной длины можно было бы определить путем разрыва гамильтонова контура перед заданным городом j_0 . Однако длина гамильтонова пути во многом зависит от его начала. Поэтому возникают данная и последующая задачи.

Для решения данной задачи в матрице затрат C заменяют все элементы столбца j_0 , соответствующего началу пути, кроме диагонального элемента, на нули, делая тем самым заезд в город j_0 беззатратным.

Решение задачи коммивояжера без заданного начала пути

Требуется определить тот город, из которого можно отправить коммивояжера для объезда всех городов, без возвращения.

Для решения заданной задачи в матрицу затрат C добавляют новую фиктивную вершину S , с элементами $c_{is} = 0, i = \overline{1, n}$ и $c_{sj} = 0, j = \overline{1, n}, c_{ss} = M$, делая тем самым заезд в город S и выезд из него беззатратными. С помощью алгоритма Литтла находится минимальный гамильтонов контур. Исключив затем вершину S , получим искомый путь.



Контрольные вопросы по главе 4

1. Дайте классификацию задач целочисленного программирования. Приведите примеры.
2. Назовите методы решения задач целочисленного программирования.
3. Можно ли получить оптимальное целочисленное решение путем округления решения задачи до целочисленности?
4. Какое ограничение называется отсечением Гомори?
5. В чем сущность метода ветвей и границ?
6. Чем отличаются математические постановки задач о назначениях и о коммивояжере?
7. Каков основной недостаток решения задачи о коммивояжере алгоритмом «ближайшего соседа»?
8. Что означает степень нулевого элемента в алгоритме Литтла?
9. Что является достаточным условием существования гамильтонова контура на графе?
10. Что нужно сделать, чтобы можно было воспользоваться алгоритмом Литтла для определения гамильтонова пути с фиксированным началом?

Глава 5

НЕЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

5.1 Особенности задач нелинейного программирования

Нелинейное программирование занимается оптимизацией моделей задач, в которых либо ограничения $q_i(x)$, либо целевая функция $Z(X)$, либо то и другое нелинейны.

Найти $\max(\min)Z = z(X)$ в области

$$\Omega: \begin{cases} q_i(X) \leq b_i, & i = \overline{1, m}; \\ X \geq 0, \end{cases}$$

где R — отношение порядка ($=, \geq, \leq$); Ω — область допустимых решений; b_i — константа, $i = \overline{1, m}$; $X = (x_1, \dots, x_n) = \{x_j\}, j = \overline{1, n}$ — план или вектор управления.

Для выяснения трудностей решения задач данного класса, порождаемых нелинейностью, сопоставим задачи линейного и нелинейного программирования. Можно указать три характерные особенности для каждого класса (табл. 5.1).

Таблица 5.1 – Задачи линейного и нелинейного программирования

Задачи ЛП	Задачи НЛП
1. Область Ω допустимых планов — выпуклое множество с конечным числом угловых точек	1. Множество Ω допустимых планов может быть невыпуклым, несвязным , иметь бесконечное число крайних точек
2. Экстремум целевая функция $z(X)$ достигает в одной из крайних точек (на границе области Ω допустимых решений)	2. Экстремум может достигаться не только на границе , но и внутри области Ω допустимых решений
продолжение на следующей странице	

Таблица 5.1 – Продолжение

Задачи ЛП	Задачи НЛП
3. Экстремальное значение $z(X)$ целевой функции является и <i>глобальным</i> значением	3. Целевая функция $z(X)$ в области Ω может иметь <i>несколько локальных</i> экстремумов

На рисунке 5.1 приводится классификация задач и методов нелинейного программирования, рекомендуемых к рассмотрению в данном курсе.



Рис. 5.1 – Классификация задач и методов нелинейного программирования

В последующих параграфах будут рассмотрены более подробно задачи условной оптимизации и методы их решения. Сейчас коротко охарактеризуем решение задач безусловной оптимизации.

Методы перебора применимы для отыскания экстремумов унимодальных целевых функций. Действие любого из методов поиска заключается в сужении области поиска экстремума (длины l_0):

- до области заданной длины $\varepsilon > 0$, проводя минимальное число измерений значений функции (методы дихотомии, золотого сечения);
- до наименьших возможных размеров l_n при заданном числе измерений n (метод Фибоначчи).

Первая формулировка целесообразна в том случае, если с каждым измерением связаны значительные затраты средств или времени, однако на поиск отпускаются неограниченные средства, которые мы все же стремимся минимизировать; вторая — когда исследователь располагает ограниченными средствами и, зная расходы, связанные с каждым измерением, стремится получить наилучший результат.

Классические методы нахождения экстремумов функций предполагают, что целевые функции — непрерывные и гладкие. Для существования точки экстремума должны выполняться необходимые и достаточные условия. Необходимыми условиями существования экстремума являются требования обращения в нуль частных производных первого порядка целевой функции по каждой из переменных. Точка, найденная из необходимых условий, называется стационарной (подозрительной на оптимальную). В качестве стационарных точек могут быть точки перегиба, седловые точки и др. Поэтому необходим учет достаточных условий нахождения экстремумов функций. Он сложен для функций многих переменных как в алгебраическом, так и в вычислительном плане. Так, в случае функции двух переменных достаточным условием существования экстремума будет положительная определенность матрицы A размером 2×2 (условие Лежандра — Клебша [12, 13]), составленной из вторых частных производных функции. Недостатком классического метода дифференциального исчисления является и то, что он дает возможность найти экстремум только в том случае, если он лежит внутри области определения функции. Если экстремум находится на границе области определения, то этот метод становится бессильным.

Методы покоординатного спуска относятся к группе приближенных методов нелинейной оптимизации и направлены на уменьшение трудностей, связанных с отысканием экстремума функции цели со сложной аналитической структурой классическими методами дифференциального исчисления. Суть этих методов заключается в продвижении от исходной точки в области определения функции к точке оптимума итеративно: в методе Гаусса — последовательно по каждой из переменных (покоординатно); в градиентных методах — одновременно по всем переменным в направлении градиента или антиградиента.

Критерием окончания итеративных процедур является равенство нулю всех частных производных целевой функции, или квадрат суммы всех частных производных целевой функции должен быть не более заданного числа ε , или разность достигнутого значения целевой функции и значения в предыдущей точке должна быть не более ε , и т. д.

5.2 Метод неопределенных множителей Лагранжа

Пусть задача состоит в отыскании плана X , доставляющего экстремальное значение целевой функции

$$\max(\min): Z = z(X) \quad (5.1)$$

при ограничениях

$$q_i(X) = b_i, \quad i = \overline{1, m}. \quad (5.2)$$

Будем считать, что функции (5.1)–(5.2) непрерывны и дважды дифференцируемые по своим аргументам. Как можно решить такую задачу условной оптимизации? Рассмотрим случай, когда число переменных равно двум и число ограничений — одному:

$$\max(\min): Z = z(x_1, x_2); \quad (5.3)$$

$$q(x_1, x_2) = b. \quad (5.4)$$

Выразим переменную x_2 в уравнении (5.4), получим выражение:

$$x_2 = \varphi(x_1). \quad (5.5)$$

Подставим его в целевую функцию (5.3). Получим

$$Z = z(x_1, \varphi(x_1)) \quad (5.6)$$

как неявную функцию от переменной x_1 . Необходимым условием существования экстремума функции (5.6), включающей исходное ограничение (5.4), является условие

$$\frac{dz(x_1, \varphi(x_1))}{dx_1} = 0, \quad (5.7)$$

$$\text{или } \frac{dz(x_1, \varphi)}{dx_1} = \frac{dZ}{dx_1} = \frac{dZ}{d\varphi} \cdot \frac{d\varphi}{dx_1} = 0. \quad (5.8)$$

Продифференцируем (5.4) по x_1 как неявную функцию:

$$\frac{dq(x_1, x_2)}{dx_1} + \frac{dq(x_1, x_2)}{dx_2} \frac{dx_2}{dx_1} = 0.$$

Так как $x_2 = \varphi(x_1)$, имеем

$$\frac{dx_2}{dx_1} = \frac{d\varphi}{dx_1} = - \frac{dq(x_1, x_2)/dx_1}{dq(x_1, x_2)/dx_2}. \quad (5.9)$$

Подставим (5.9) в (5.8), получим

$$\frac{dZ}{dx_1} = \frac{dZ}{dx_1} - \frac{dZ}{dx_2} \cdot \frac{dq/dx_1}{dq/dx_2} = 0. \quad (5.10)$$

Обозначив

$$\lambda = \frac{dZ/dx_2}{dq/dx_2}, \quad (5.11)$$

получим из (5.10), (5.11), (5.4) систему уравнений:

$$\begin{cases} \frac{dZ}{dx_1} - \lambda \frac{dq}{dx_1} = 0; \\ \frac{dZ}{dx_2} - \lambda \frac{dq}{dx_2} = 0; \\ q(x_1, x_2) = b. \end{cases} \quad (5.12)$$

Система уравнений (5.12) есть необходимое условие существования условного локального экстремума задачи (5.3)–(5.4). Решив систему (5.12) с неизвестными x_1 , x_2 и λ , найдем все точки X^* , подозрительные на экстремум. Систему (5.12) необходимых условий определения подозрительных точек на экстремум X^* можно получить формальным путем. Составим вспомогательную функцию $L(X, \lambda) = z(x_1, x_2) + \lambda[b - q(x_1, x_2)]$. Она называется функцией Лагранжа, а λ — неопределенным множителем Лагранжа. Если в ней считать x_1 , x_2 и λ независимыми переменными, найти частные производные по x_1 , x_2 и λ и приравнять их к нулю, то получим систему (5.12).

Для задачи в общем виде (5.1)–(5.2) функция Лагранжа будет иметь вид:

$$L(X, \Lambda) = z(X) + \sum_{i=1}^m \lambda_i [b_i - q_i(X)],$$

где $(x_1, \dots, x_n)$, $\Lambda = (\lambda_1, \dots, \lambda_m)$.

Исходная задача (5.1)–(5.2) отыскания условного экстремума заменяется задачей отыскания безусловного экстремума функции Лагранжа $L(X, \Lambda)$, которая может быть решена через систему уравнений:

$$\begin{cases} \frac{dL(X, \Lambda)}{dx_j} = \frac{dz(X)}{dx_j} - \sum_{i=1}^m \lambda_i \frac{dq_i(X)}{dx_j} = 0, \quad j = \overline{1, n}; & (a) \\ \frac{dL(X, \Lambda)}{d\lambda_i} = b_i - q_i(X) = 0, \quad i = \overline{1, m}. & (б) \end{cases} \quad (5.13)$$

Условия (5.13, а)–(5.13, б) необходимы для экстремума задачи (5.1)–(5.2), т. е. каждая точка X^* задачи (5.1)–(5.2) является решением условий (5.13, а)–(5.13, б). Однако условия (5.13, а)–(5.13, б) не являются достаточными, т. е. не обязательно, чтобы любое решение системы (5.13, а)–(5.13, б) доставляло экстремальное значение функции (5.1).



Пример 5.1

Исследовать точки на экстремум $Z = x_1^2 + x_2^2$; $x_1 + x_2 = 1$.

Решение:

Составим функцию Лагранжа:

$$L(x_1, x_2, \lambda) = x_1^2 + x_2^2 + \lambda(1 - x_1 - x_2).$$

Составим необходимые условия существования экстремума:

$$\begin{cases} \frac{dL}{dx_1} = 2x_1 - \lambda = 0; \\ \frac{dL}{dx_2} = 2x_2 - \lambda = 0; \\ \frac{dL}{d\lambda} = 1 - x_1 - x_2 = 0. \end{cases}$$

Решим систему уравнений $2x_1 - \lambda = 0$, $2(x_1 + x_2) = 2\lambda$, а т. к. $x_1 + x_2 = 1$, то $\lambda = 1$, тогда $x_1 = 1/2$, $x_2 = 1/2$. Значение целевой $Z = (1/2)^2 + (1/2)^2 = 1/2$.

Чтобы оценить, является ли точка экстремальной и какой экстремум она дает, обратимся к достаточному условию существования экстремума функции двух переменных (условию Лежандра — Клебша [12, 13]).

Составим определитель:

$$\Delta = \begin{vmatrix} \frac{d^2 Z(X^*)}{dx_1^2} & \frac{d^2 Z(X^*)}{dx_1 dx_2} \\ \frac{d^2 Z(X^*)}{dx_1 dx_2} & \frac{d^2 Z(X^*)}{dx_2^2} \end{vmatrix} = \begin{vmatrix} 2 & 0 \\ 0 & 2 \end{vmatrix} > 0.$$

Так как $d^2 Z(X^*)/dx_1^2 > 0$ и $\Delta > 0$, то в точке $X^*(1/2, 1/2)$ функция Z достигает минимального значения. Графическое решение (рис. 5.2) показывает, что максимальное значение $Z_{\max} = 1$ достигается в точках $(x_1 = 1; x_2 = 0)$ и $(x_1 = 0; x_2 = 1)$.

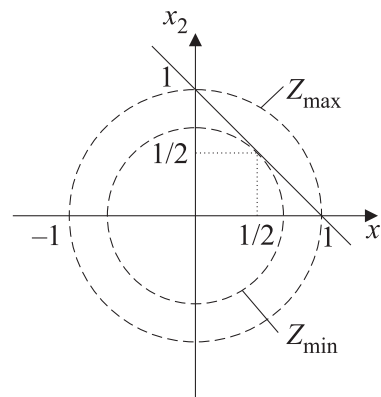


Рис. 5.2 – Графическое решение

Таким образом, если бы исходная задача в примере ставилась на отыскание максимума, то с помощью решения системы уравнений необходимых условий существования экстремума мы точку максимума бы не определили. Требуется иной подход, который рассмотрим ниже.

5.3 Функция Лагранжа для задачи линейного программирования. Понятие седловой точки функции Лагранжа

Возьмем пару двойственных задач ЛП.

Прямая задача $\max : z(X) = \sum_{j=1}^n c_j x_j,$ $\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = \overline{1, m}$ $x_j \geq 0, \quad j = \overline{1, n}.$	y_i	Двойственная задача $\min : f(Y) = \sum_{i=1}^m b_i y_i$ $\sum_{i=1}^m a_{ij} y_i \geq c_j, \quad j = \overline{1, n}$ $y_i \geq 0, \quad i = \overline{1, m}$
---	-------	---

Запишем в другом виде:

$$\left. \begin{array}{l} \max : z(X) = \sum_{j=1}^n c_j x_j, \\ b_i - \sum_{j=1}^n a_{ij} x_j \geq 0, \quad i = \overline{1, m}, \\ x_j \geq 0, \quad j = \overline{1, n}, \end{array} \right| y_i \left| \begin{array}{l} \max : -f(Y) = -\sum_{i=1}^m b_i y_i, \\ \sum_{i=1}^m a_{ij} y_i - c_j \geq 0, \quad j = \overline{1, n}, \\ y_i \geq 0, \quad i = \overline{1, m}, \end{array} \right| x_j,$$

где y_i — двойственная оценка для прямой задачи; x_j — двойственная оценка для двойственной задачи.

Распространим функцию Лагранжа на задачи, в которых ограничения не обязательно имеют форму равенств. Запишем функции Лагранжа для прямой и двойственной задач:

$$L_z(X, Y) = \sum_{j=1}^n c_j x_j + \sum_{i=1}^m y_i \left(b_i - \sum_{j=1}^n a_{ij} x_j \right); \quad (5.14)$$

$$L_f(Y, X) = \sum_{i=1}^m b_i y_i + \sum_{j=1}^n y_j \left(\sum_{i=1}^m a_{ij} y_i - c_j \right). \quad (5.15)$$

Из (5.14)–(5.15) следует, что для прямой ЗЛП двойственные оценки y_i служат множителями Лагранжа, для двойственной ЗЛП множителями Лагранжа являются переменные x_j . Сложив почленно (5.14) и (5.15), получим $L_z(X, Y) = -L_f(Y, X)$.



.....
Функция $L(X, Y)$ имеет в точке (X°, Y°) **седловую точку**, если

$$L(X, Y^\circ) \leq L(X^\circ, Y^\circ) \leq L(X^\circ, Y), \quad (5.16)$$

для всех X из $\varepsilon > 0$ — окрестности X° и всех Y из $\varepsilon > 0$ — окрестности Y° .
.....

Если неравенства (5.16) выполняются для всех X и Y из области определения функции $L(X, Y)$, то она имеет глобальную седловую точку (X°, Y°) . Оптимальные решения пары двойственных задач ЛП образуют седловую точку функции Лагранжа (5.14). Геометрическая интерпретация седловой точки функции Лагранжа показана на рисунке 5.3.

Существует следующая теорема [1, 12].



.....
Чтобы план X° был оптимальным решением прямой задачи ЛП, необходимо и достаточно, чтобы существовал такой вектор $Y^\circ \geq 0$, когда точка (X°, Y°) была бы седловой точкой функции Лагранжа $L_z(X, Y)$, т. е. для всех $X \geq 0$ и $Y \geq 0$:

$$L_z(X, Y^\circ) \leq L(X^\circ, Y^\circ) \leq L(X^\circ, Y);$$

$$L_z(X^\circ, Y^\circ) = \max_{X \geq 0} L_z(X, Y^\circ);$$

$$L_z(X^\circ, Y^\circ) = \min_{Y \geq 0} L_z(X^\circ, Y);$$

$$L_z(X^\circ, Y^\circ) = \max_{X \geq 0} \min_{Y \geq 0} L_z(X, Y) = \min_{Y \geq 0} \max_{X \geq 0} L_z(X, Y).$$

.....

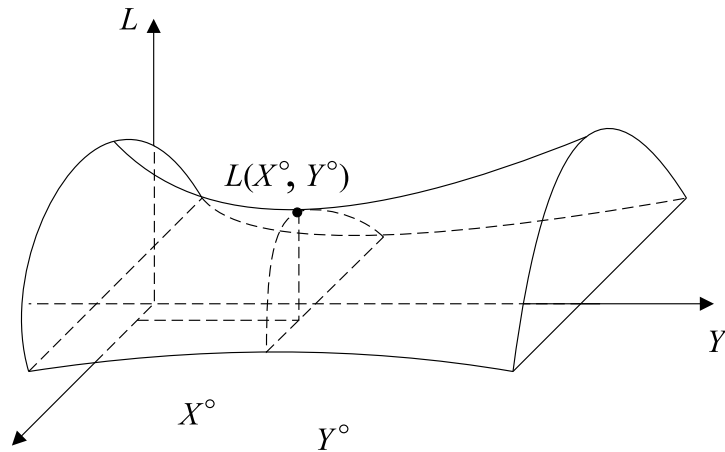


Рис. 5.3 – Геометрическая интерпретация седловой точки

Задача определения оптимальных решений пары двойственных задач сводится к нахождению максимина или минимакса функции Лагранжа.

5.4 Теорема Куна — Таккера

Пусть дана задача нелинейного программирования:

$$\max : Z = z(X), \quad (5.17)$$

$$q_i(X) \leq b_i, \quad i = \overline{1, m}; \quad (5.18)$$

$$x_j \geq 0, \quad j = \overline{1, n}. \quad (5.19)$$

Запишем функцию Лагранжа:

$$L(X, \Lambda) = z(X) + \sum_{i=1}^m \lambda_i (b_i - q_i(X)), \quad (5.20)$$

где $\lambda_i, i = \overline{1, m}$ — неопределенные множители Лагранжа; $z(X)$ и $q_i(X)$ — выпуклые вверх функции.



.....
Теорема Куна — Таккера. Чтобы план X° был решением задачи (5.16)–(5.19), необходимо и достаточно, чтобы существовал вектор $\Lambda^\circ \geq 0$, такой, что пара (X°, Λ°) для всех $X \geq 0$ и $\Lambda \geq 0$.

$$L(X, \Lambda^\circ) \leq L(X^\circ, \Lambda^\circ) \leq L(X^\circ, \Lambda).$$

Чтобы функция двух векторных переменных (5.20) имела седловую точку, необходимо и достаточно выполнения следующих условий:

$$\frac{dL(X^\circ, \Lambda^\circ)}{dx_j} \leq 0; \quad (5.21)$$

$$x_j^\circ \frac{dL(X^\circ, \Lambda^\circ)}{dx_j} = 0, \quad x_j^\circ \geq 0; \quad (5.22)$$

$$\frac{dL(X^\circ, \Lambda^\circ)}{d\lambda_i} \geq 0; \quad (5.23)$$

$$\lambda_i^\circ \frac{dL(X^\circ, \Lambda^\circ)}{d\lambda_i} = 0, \quad \lambda_i^\circ \geq 0. \quad (5.24)$$

.....

Если решается задача **минимизации**, то имеет место седловая точка (X°, Y°) , если выполняются соотношения:

$$L(X, \Lambda^\circ) \geq L(X^\circ, Y^\circ) \geq L(X^\circ, Y).$$

Условия же Куна — Таккера существования седловой точки функции Лагранжа поменяют знак в (5.21) и (5.23) на противоположный.

По содержанию условия Куна — Таккера (5.21)–(5.24) эквивалентны условиям второй теоремы двойственности задач линейного программирования.

5.5 Метод линейной аппроксимации



.....
Функция n переменных $f(x_1, x_2, \dots, x_n)$ называется **сепарабельной**, если ее можно представить как сумму n функций от одной переменной x_j , $j = \overline{1, n}$, т. е.

$$f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n f_i(x_j).$$

.....

Задачи нелинейного программирования, у которых функция цели и функции системы ограничений сепарабельны, называются задачами **сепарабельного программирования**.

Задача сепарабельного программирования примет вид:

$$\max(\min): Z = \sum_{j=1}^n z_j(x_j), \quad (5.25)$$

при ограничениях

$$\sum_{j=1}^n g_{ij}(x_j) R b_i, \quad i = \overline{1, m}; \quad (5.26)$$

$$x_j \geq 0, \quad j = \overline{1, n}, \quad (5.27)$$

где R — отношение $\leq$, $\geq$ или $=$.

Очевидно, данную задачу можно решить общими методами нелинейного программирования. Однако с учетом специфики разработан специальный приближенный метод, позволяющий свести задачу к линейному программированию. Он основан на кусочно-линейной аппроксимации (приближении) функций $z_j(x_j)$ и $g_{ij}(x_j)$. Любая непрерывная функция $h(x)$ может быть аппроксимирована с наперед заданной степенью точности кусочно-линейной функцией $\bar{h}(x)$ (рис. 5.4). Этот факт положен в основу линейной аппроксимации задач с сепарабельными функциями.

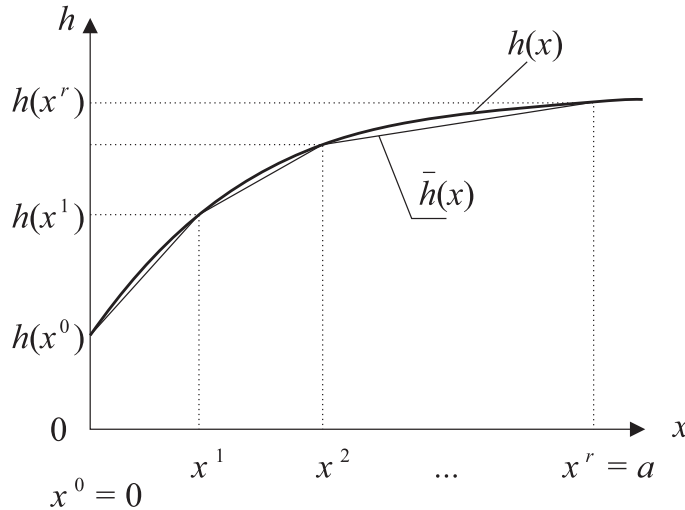


Рис. 5.4 – Кусочно-линейная функция

Пусть задана непрерывная функция одной переменной $h(x)$, определенная на отрезке $[0, a]$. Разобьем отрезок $[0, a]$ на r частей, получим точки $x^0 = 0, x^1, x^2, \dots, x^r = a : x^k, k = \overline{0, r}$ со значениями функций $h(x^k), k = \overline{0, r}$.

Тогда непрерывная функция может быть аппроксимирована на отрезке $[0, a]$ аналитическими выражениями [12]:

$$x = \sum_{k=0}^r \lambda^k x^k;$$

$$\bar{h}(x) = \sum_{k=0}^r \lambda^k \bar{h}^k;$$

$$\sum_{k=0}^r \lambda^k = 1, \lambda^k \geq 0, k = \overline{1, r}.$$

Запишем аналитические выражения аппроксимирующих функций исходной задачи (5.25)–(5.27).

Пусть $a_j = \max x_j$, т. е. максимальное значение, которое может принимать переменная x_j . Разобьем каждый из отрезков $[0, a_j] j = \overline{1, n}$ на r_j частей с помощью точек x_j^k , где $x_j^0 = 0, x_j^{r_j} = a_j$.

Задача (5.25)–(5.27) запишется так:

$$\max(\min): \bar{Z} = \sum_{j=1}^n \sum_{k=0}^{r_j} \lambda_j^k z_j^k, \quad (5.28)$$

при ограничениях

$$\sum_{j=1}^n \sum_{k=0}^{r_j} \lambda_j^k g_{ij}^k Rb_i, \quad i = \overline{1, m}; \quad (5.29)$$

$$\sum_{k=0}^{r_j} \lambda_j^k = 1, \quad j = \overline{1, n}; \quad (5.30)$$

$$\lambda_j^k \geq 0, \quad k = \overline{1, r_j}, j = \overline{1, n}. \quad (5.31)$$

Задача (5.28)–(5.31) относительно переменных λ_j^k называется аппроксимирующей для (5.25)–(5.27). Если λ_j^{k*} — оптимальное решение аппроксимирующей задачи (5.28)–(5.31), тогда оптимальное решение исходной можно рассчитать по формуле:

$$x_j^* = \sum_{k=0}^{r_j} \lambda_j^{k*} x_j^k, \quad j = \overline{1, n}. \quad (5.32)$$

Если некоторая переменная x_j входит в исходную задачу линейно, то ее не следует выражать через λ_j^k , а просто использовать как переменную в аппроксимирующей задаче.



Пример 5.2

$$\begin{aligned} \max : Z &= 4x_1 - 2x_1^2 + 2x_2; \\ 3x_1^2 + 4x_2^2 &\leq 12; \\ x_1 &\geq 0; \quad x_2 \geq 0. \end{aligned}$$

Введем обозначения:

$$\begin{aligned} z_1 &= 4x_1 - 2x_1^2; \quad z_2 = 2x_2; \\ g_{11} &= 3x_1^2; \quad g_{12} = 4x_2^2. \end{aligned}$$

Из ограничения определяем максимальные значения переменных и область их изменения: $0 \leq x_1 \leq 2$; $0 \leq x_2 \leq \sqrt{3} \approx 2$. Возьмем число участков аппроксимации $r_j = 2, j = \overline{1, 2}$. Определим значения z_j^k и g_{1j}^k в точках

$$x_j^k, \quad k = 0, 1, 2; \quad j = 1, 2 : x_1^0 = 0, x_1^1 = 1, x_1^2 = 2, x_2^0 = 0, x_2^1 = 1, x_2^2 = 2.$$

Результаты сведем в таблицу 5.2.

Таблица 5.2 – Данные аппроксимации функций

k	x_1^k	x_2^k	z_1^k	z_2^k	g_{11}^k	g_{12}^k
0	0	0	0	0	0	0
1	1	1	2	2	3	4
2	2	2	0	4	12	16

Аппроксимирующая задача линейного программирования примет вид:

$$\begin{aligned} \max : \bar{Z} &= 2\lambda_1^1 - 2\lambda_2^1 + 4\lambda_2^2; \\ 3\lambda_1^1 + 12\lambda_1^2 + 4\lambda_2^1 + 16\lambda_2^2 &\leq 12; \\ \lambda_1^0 + \lambda_1^1 + \lambda_1^2 &= 1; \quad \lambda_2^0 + \lambda_2^1 + \lambda_2^2 = 1; \\ \lambda_j^k &\geq 0; \quad k = 0, 1, 2; \quad j = 1, 2. \end{aligned}$$

Решив ее симплексным методом, получим

$$\lambda_1^{1*} = 1; \lambda_1^{2*} = 0; \lambda_2^{1*} = \frac{7}{12}; \lambda_2^{2*} = \frac{5}{12}.$$

Используя (5.32), запишем решение исходной задачи:

$$x_1^* = \sum_{k=0}^2 \lambda_1^{k*} x_1^k = 0 \cdot 0 + 1 \cdot 1 + 0 \cdot 2 = 1;$$

$$x_2^* = \sum_{k=0}^2 \lambda_2^{k*} x_2^k = \frac{17}{12};$$

$$Z = \sum_{j=1}^2 \sum_{k=0}^2 \lambda_2^{k*} z_j^k = \frac{29}{6}.$$



Контрольные вопросы по главе 5

1. В чем отличительные особенности задач нелинейного программирования от задач ЛП?
2. Дайте классификацию задач нелинейного программирования.
3. Что является необходимым условием существования экстремума для задачи безусловной оптимизации?
4. Дайте понятие седловой точки функции Лагранжа для задачи линейного программирования.
5. Запишите условия существования седловой точки функции Лагранжа (условия Куна — Таккера).
6. Покажите связь теорем Куна — Таккера для задачи нелинейного программирования и двойственности для задачи линейного программирования.
7. В чем суть метода линейной аппроксимации?
8. Чем отличается задача линейного программирования от задачи квадратического программирования?
9. Почему для решения задачи квадратического программирования можно применить симплекс-метод?
10. Опишите алгоритм решения задачи квадратического программирования на базе теоремы Куна — Таккера.

Глава 6

ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

6.1 Особенности задач динамического программирования

Динамическое программирование (ДП) представляет собой динамический аппарат, разработанный с целью повышения эффективности вычислений при решении некоторого класса задач математического программирования путем их разложения на относительно небольшие. Характерным для ДП является подход к решению задачи по этапам, с каждым из которых ассоциирована одна управляемая переменная. Набор рекуррентных вычислительных процедур, связывающих различные этапы, обеспечивает получение оптимального решения задачи в целом при достижении последнего этапа.

Возникновение динамического программирования связано с именем американского ученого Р. Беллмана, который в начале 1950-х годов применил к ряду конкретных задач прием, названный впоследствии принципом оптимальности, заключающийся в последовательном (поэтапном) анализе процесса. Трудно дать четкое описание процесса решения задач динамического программирования, укажем лишь на его характерные особенности.

1. Рассматривается система S (см. рис. 6.1), состояние которой на каждом шаге (этапе) $t (t = 1, N)$ определяется вектором $S_t = (s_{t1}, s_{t2}, \dots, s_{ti}, \dots, s_{tm})$, где s_{ti} — одно из возможных состояний системы на шаге t . Дальнейшее изменение ее состояния зависит только от данного состояния S_t и не зависит от того, каким путем система пришла в него. Такие процессы называются **марковскими** процессами без последствия.

2. На каждом шаге выбирается одно решение из вектора $X_t = (x_{t1}, x_{t2}, \dots, x_{ti}, \dots, x_{tm})$, под действием которого система переходит из предыдущего состояния S_{t-1} в новое S_t , где x_{ti} — одно из возможных решений на этапе t . Очевидно, оно будет функцией $S_t = S_t(S_{t-1}, X_t)$.

3. Выбор решения x_{it} на каждом шаге t связан с целевой функцией управления на данном шаге z_t . Состояние системы S_t на каждом шаге может быть оценено через целевую функцию эффективности (или потерь) системы $F_t(S_t)$, равную сумме целевых функций управления этапов:

$$F_t(S_t) = \sum_{k=1}^t z_k.$$

4. На векторы состояния S_t и управления X_t накладываются ограничения, объединение которых составляет область допустимых решений Ω задачи ДП:

$$(S_t \cap X_t) \in \Omega.$$

5. Требуется найти такое допустимое управление X_t для каждого шага $t = \overline{1, N}$, чтобы получить экстремальное значение функции цели системы $F_N(S_N)$ за N шагов.

Любую допустимую последовательность действий для каждого шага, переводящую систему из начального состояния S_0 в конечное S_N , называют **стратегией управления**. Допустимая стратегия управления, доставляющая функции цели F_N экстремальное значение, называется **оптимальной**.

Вычислительная схема метода динамического программирования предусматривает анализ этапов дважды. Первый раз — от начала к концу (по алгоритму прямой прогонки). Генерируются все возможные траектории из начального состояния S_0 в конечное S_N .

На каждом шаге движения ищется условно оптимальное управление и экстремальное значение функции цели для процесса. Второй раз — от конца к началу. Из сгенерируемых траекторий на каждом шаге от S_N к S_0 выбирается наилучшая, определяя оптимальное управление с точки зрения всего процесса. Проинтерпретировать вычислительную схему метода ДП можно на следующем примере.

Из исходного пункта группе путников предлагается пройти путь своим маршрутом в конечный пункт за минимальное время и с отметкой о прохождении в промежуточных пунктах. По прибытии в конечный пункт путник, пришедший с наилучшим временем, двигаясь обратно, должен показать свой маршрут.

6.2 Принципы динамического программирования. Функциональные уравнения Беллмана

Прежде чем привести формулировку принципов ДП, остановимся на решении следующей задачи — распределения однородного ресурса между предприятиями.

Пусть в объединении имеется три предприятия ($j = \overline{1, 3}$), эффективность работы $z_j x_j$ каждого зависит от выделенного ресурса в объеме x_j (табл. 6.1). Всего объединение располагает ресурсом в объеме b , равном двум единицам. Необходимо распределить ограниченный ресурс между предприятиями, чтобы суммарная эффективность работы объединения была максимальной. Для простоты изложения будем считать, что ресурс распределяется дискретно.

Составим математическую модель задачи.

Таблица 6.1 – Эффективность работы предприятий

Объем ресурса x	Эффективность $z_j(x)$		
	Предприятие 1	Предприятие 2	Предприятие 3
0	0	0	0
1	5	6	7
2	8	10	13

Максимизировать

$$F = \sum_{j=1}^3 z_j(x_j), \quad (6.1)$$

при ограничениях

$$\sum_{j=1}^3 x_j = b; \quad (6.2)$$

$$x_j \geq 0, j = \overline{1, 3}; \quad (6.3)$$

$$x_j — \text{целые}, j = \overline{1, 3}. \quad (6.4)$$

В общем виде $z_j(x_j)$ для каждого предприятия может быть задана аналитически, графически, таблично и т. п. Если бы $z_j(x_j)$ была задана в виде линейной или нелинейной функции, то можно было бы применить методы линейного или нелинейного программирования для ее решения. В любом случае ее можно решить методом динамического программирования, если прослеживаются упомянутые выше особенности ДП.

1. Состояние системы распределения ресурса свяжем с ограничением (6.2):

$$S_1 = x_1;$$

$$S_2 = x_1 + x_2;$$

$$S_3 = x_1 + x_2 + x_3.$$

В общем виде состояние системы S_j на этапе j связано с распределением ресурса между первым, вторым, ..., j -м предприятиями:

$$S_j = \sum_{t=1}^j x_t, j = \overline{1, 3}.$$

2. Управляемая переменная x_j — объем выделенного ресурса предприятию j — на каждом этапе j не зависит от предыдущего состояния S_{j-1} , а влияет на новое состояние $S_j, j = \overline{1, 3}$.
3. Цель управления на каждом этапе j определяется целевой функцией предприятия $z_j(x_j)$, целевая функция системы на этапе j определяется следующим образом:

$$F_j(S_j) = \sum_{t=1}^j z_t(x_t), j = \overline{1, 3}.$$

4. На состояние системы S_j и значения управляемых переменных x_j на этапах j наложены ограничения:

$$S_0 = 0; \quad 0 \leq S_1 \leq b; \quad 0 \leq S_2 \leq b; \quad S_3 = b;$$

$$x_1 = 0, 1, 2; \quad x_2 = 0, 1, 2; \quad x_3 = 0, 1, 2.$$

Вместо изучения одной задачи (6.1)–(6.4) рассмотрим семейство таких задач: максимизировать

$$F_t(S_t) = \sum_{j=1}^t z_j(x_j), \quad (6.5)$$

при ограничениях

$$\sum_{j=1}^t x_j = S_t; \quad (6.6)$$

$$x_j \geq 0, \quad j = \overline{1, t}; \quad (6.7)$$

$$x_j - \text{целые}, \quad j = \overline{1, t}; \quad (6.8)$$

$$0 \leq S_t \leq b, \quad t = 1, 2, 3. \quad (6.9)$$

Таким образом, решение исходной задачи (6.1)–(6.4) мы искусственно разворачиваем во времени, представляя как N -шаговый процесс ($N = 3$). Состояние процесса определяется величиной S_t на шаге t , а эффективность состояния системы на шаге t определяется функцией $F_t(S_t)$. Процесс возможных изменений состояния системы можно представить в виде графа состояний (рис. 6.1).

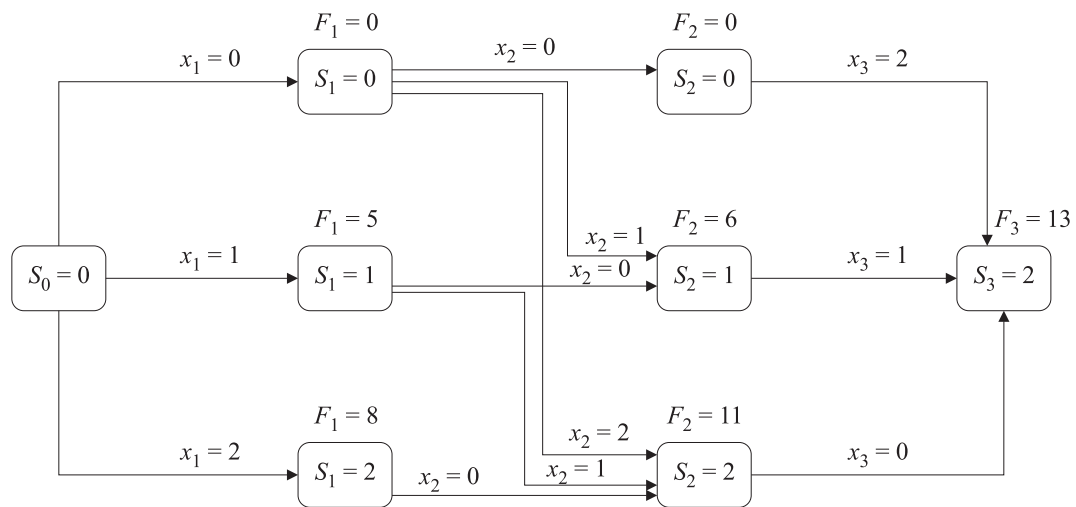


Рис. 6.1 – Граф возможных состояний системы

Эффективность состояния системы S_0 на нулевом шаге $F_0(S_0)$ равна нулю. На первом шаге управляемая переменная x_1 — объем ресурса, выделяемого только первому предприятию, может принимать значения $x_1 = 0, x_1 = 1, x_1 = 2$. Соответственно, состояние системы на первом шаге $S_1 = x_1$ может принимать значения $S_1 = 0, S_1 = 1, S_1 = 2$.

Модель (6.5)–(6.9) будет иметь вид:

$$F_1(S_1) = \max_{\substack{0 \leq x_1 \leq 2 \\ x_1 = S_1}} z_1(x_1).$$

На рисунке 6.1 значения $F_t(S_t)$ проставлены над вершинами графа. Из таблицы 6.1:

$$F_1(S_1 = 0) = z_1(x_1 = 0) = 0; F_1(S_1 = 1) = z_1(x_1 = 1) = 5; F_1(S_1 = 2) = z_1(x_1 = 2) = 8.$$

Поместим значения функции $F_1(S_1)$ в таблицу 6.2.

Таблица 6.2 – Условно оптимальные решения на этапе 1

Состояние S_1	$F_1(S_1) = z_1(x_1)$	<i>opt</i>	
		$F_1^*(S_1)$	x_1^*
0	0	0	0
1	5	5	1
2	8	8	2

Предположим теперь, что ресурс $b = 2$ единицам распределяется между двумя предприятиями. Состояние системы S_2 на втором этапе будет определяться суммой возможных значений управляемых переменных x_1 и x_2 :

$$S_2 = x_1 + x_2.$$

Модель (6.5)–(6.9) будет иметь вид:

$$\begin{aligned}
 F_2(S_2) &= \max_{\substack{0 \leq x_1 \leq 2 \\ 0 \leq x_2 \leq 2 \\ x_1 + x_2 = S_2 \\ 0 \leq S_2 \leq 2}} \{z_1(x_1) + z_2(x_2)\} = \max_{\substack{0 \leq x_2 \leq 2 \\ 0 \leq S_2 \leq 2}} \left\{ \max_{\substack{0 \leq x_1 \leq 2 \\ x_1 = S_2 - x_2}} \{z_2(x_2) + z_1(x_1)\} \right\} = \\
 &= \max_{\substack{0 \leq x_2 \leq 2 \\ 0 \leq S_2 \leq 2}} \left\{ z_2(x_2) + \max_{\substack{0 \leq x_1 \leq 2 \\ x_1 = S_2 - x_2 = S_1}} z_1(x_1) \right\} = \max_{\substack{0 \leq x_2 \leq 2 \\ 0 \leq S_2 \leq 2}} \{z_2(x_2) + F_1(S_1)\} = \\
 &= \max_{\substack{0 \leq x_2 \leq 2 \\ 0 \leq S_2 \leq 2 \\ x_2 \leq S_2}} \{z_2(x_2) + F_1(S_2 - x_2)\}.
 \end{aligned}$$

Определим условные оптимумы $F_2(S_2)$ при различных значениях S_2 .

$$\text{При } S_2 = 0: F_2(0) = \max_{x_2=0} \{z_2(0) + F_1(0)\} = 0.$$

$$\text{При } S_2 = 1: F_2(1) = \max_{x_2=0,1} \{z_2(0) + F_1(1); z_2(1) + F_1(0)\} = \max\{0 + 5; 6 + 0\} = 6.$$

$$\begin{aligned}
 \text{При } S_2 = 2: F_2(2) &= \max_{x_2=0,1,2} \{z_2(0) + F_1(2); z_2(1) + F_1(1); z_2(2) + F_1(0)\} = \\
 &= \max\{0 + 8; 6 + 5; 10 + 0\} = 11.
 \end{aligned}$$

Вычисления $F_2(S_2)$ сведем в таблицу 6.3.

Таблица 6.3 – Условно оптимальные решения на этапе 2

Состояние S_2	$z_2(x_2) + F_1(S_2 - x_2)$			<i>opt</i>	
	$x_2 = 0$	$x_2 = 1$	$x_2 = 2$	$F_2^*(S_2)$	x_2^*
0	0 + 0	–	–	0	0
1	0 + 5	6 + 0	–	6	1
2	0 + 8	6 + 5	10 + 0	11	1

Значения $F_2(x_2)$ берутся из таблицы 6.1, а $F_1(S_2 - x_2)$ – из таблицы 6.2.

На третьем шаге $S_3 = x_1 + x_2 + x_3 = 2$. Модель (6.5)–(6.9) будет иметь вид:

$$F_3(S_3) = \max_{\substack{x_1, x_2, x_3 \\ x_1 + x_2 + x_3 = S_3}} \{z_1(x_1) + z_2(x_2) + z_3(x_3)\} = \max_{\substack{0 \leq x_3 \leq 2 \\ S_3 = 2 \\ x_3 \leq S_3}} \{z_3(x_3) + F_2(S_3 - x_3)\}.$$

На последнем этапе определяется оптимальное (наилучшее) значение целевой функции $F_3(S_3)$ при $S_3 = 2$.

$$F_3(2) = \max_{x_3=0,1,2} \{z_3(0) + F_2(2); z_3(1) + F_2(1); z_3(2) + F_2(0)\} = \{0 + 11; 7 + 6; 13 + 0\} = 13.$$

Вычисления $F_3(S_3)$ сведем в таблицу 6.4.

Таблица 6.4 – Условно оптимальные решения на этапе 3

Состояние S_3	$z_3(x_3) + F_2(S_3 - x_3)$			<i>opt</i>	
	$x_3 = 0$	$x_3 = 1$	$x_3 = 2$	$F_3^*(S_3)$	x_3^*
2	0 + 11	7 + 6	13 + 0	13	1 или 2

Максимальная суммарная эффективность работы предприятий равна 13 единицам.

Определение оптимальных значений управляемых переменных x_j (табл. 6.5) начинается с последней таблицы (обратный ход).

Таблица 6.5 – Определение управляемых переменных

	Из табл. 6.4	$S_2 = S_3 - x_3$	Из табл. 6.3	$S_1 = S_2 - x_2$	Из табл. 6.2
Вариант 1	$x_3^* = 1$	$S_2 = 2 - 1 = 1$	$x_2^* = 1$	$S_1 = 1 - 1 = 0$	$x_1^* = 0$
Вариант 2	$x_3^* = 2$	$S_2 = 2 - 2 = 0$	$x_2^* = 0$	$S_1 = 0 - 0 = 0$	$x_1^* = 0$

Оптимальные решения: $x_1^* = (0, 1, 1)$ и $x_2^* = (0, 0, 2)$.

В общем виде эффективности возможных состояний системы (условные оптимумы) определяются через функциональные уравнения, получившие название **уравнений Беллмана**:

$$F_0(S_0) = 0;$$

$$F_t(S_t) = \max(\min_{x_t, S_t}) \{z_t(x_t) + F_{t-1}(S_{t-1})\}, \quad t = \overline{1, N},$$

в которых состояние S_{t-1} выражается через x_t и S_t .

Из качественного анализа идеи поэтапной оптимизации можно сформулировать следующие принципы, лежащие в основе динамического программирования:

- 1) **принцип оптимальности.** Оптимальное уравнение на каждом шаге определяется состоянием системы на начало этого шага и целью управления;
- 2) **принцип инвариантного погружения** (вложения). Форма задачи ДП инвариантна относительно количества шагов.

Применение принципов оптимальности и погружения в многошаговых, дискретных процессах приводит к рекуррентно-функциональным уравнениям, позволяющим последовательно выписать оптимальные уравнения для исходной задачи. Выигрыш здесь состоит в том, что задача вычисления для всего процесса управления разбивается на ряд более простых задач для отдельных этапов процесса.

6.3 Задача о рюкзаке

Рассмотрим решение задачи о рюкзаке с помощью уравнений Р. Беллмана. Пример взят из [1].



Пример 6.1

Самолет загружается предметами N различных типов (табл. 6.6) с весом w_j и стоимостью c_j , $j = \overline{1, n}$. Максимальная грузоподъемность равна $W = 5$. Определить максимальную стоимость груза, вес которого не более W .

Таблица 6.6 – Исходные данные

Тип j	Вес w_j	Стоимость c_j
1	2	65
2	3	80
3	1	30

Решение:

Сделаем математическую постановку. Пусть x_j — количество предметов j -го типа, загружаемых в самолет. Тогда математическая модель имеет вид:

$$Z(X) = \sum_{j=1}^n c_j x_j \rightarrow \max, \quad (6.10)$$

$$\sum_{j=1}^n w_j x_j \leq W, \quad (6.11)$$

$$x_j \geq 0, j = \overline{1, n}, \quad (6.12)$$

$$x_j — \text{целые}. \quad (6.13)$$

Отличительные особенности задачи динамического программирования:

1. Этап j связан с загрузкой предметов j -го типа в количестве x_j единиц (x_j — управляемая переменная).
2. Состояние загружаемого самолета S_j на этапе j определяется через ограничение (6.11) математической модели (6.10)–(6.13). В алгоритме прямой прогонки состояния:

$$S_j = \sum_{i=1}^j w_i x_i, j = \overline{1, n}; 0 \leq S_j \leq W; S_n = W.$$

В алгоритме обратной прогонки:

$$S_j = \sum_{i=j}^n w_i x_i, j = \overline{1, n}; 0 \leq S_j \leq W; S_1 = W.$$

3. Цель управления на этапе $z_j = c_j x_j, j = \overline{1, n}$.
4. Варианты решения x_j этапа j описываются количеством предметов типа j :

$$x_j = 0, 1, \dots, [W/w_j] — \text{целые}.$$

Решим задачу методом обратной прогонки, загружая предметы с последнего типа.

Пусть $F_j(S_j)$ — значение целевой функции: максимальная стоимость предметов, включенных на этапах $j, j = 1, \dots, n$ при заданном состоянии системы S_j .

Рекуррентное соотношение для процедур обратной прогонки:

$$\begin{aligned} F_{n+1}(S_{n+1}) &= 0; \\ F_j(S_j) &= \max\{c_j x_j + F_{j+1}(S_j - w_j x_j)\}, j = 1, \dots, n; \\ S_j &= 0, 1, \dots, W; \\ S_j &\geq w_j x_j. \end{aligned}$$

Этап 1.

$$\begin{aligned} F_3(S_3) &= \max\{c_3 x_3\}; \\ x_3 &= 0, 1, \dots, 5; \\ S_3 &= 0, 1, \dots, 5; \\ S_3 &\geq 1x_3. \end{aligned}$$

S_3	c_3x_3						opt	
	$x_3 = 0$	$x_3 = 1$	$x_3 = 2$	$x_3 = 3$	$x_3 = 4$	$x_3 = 5$	$F_3^*(S_3)$	x_3^*
0	0	–	–	–	–	–	0	0
1	0	30	–	–	–	–	30	1
2	0	30	60	–	–	–	60	2
3	0	30	60	90	–	–	90	3
4	0	30	60	90	120	–	120	4
5	0	30	60	90	120	150	150	5

Этап 2.

$$F_2(S_2) = \max\{c_2x_2 + F_3(S_2 - w_2x_2)\};$$

$$x_2 = 0, 1;$$

$$S_2 = 0, 1, \dots, 5; S_2 \geq 3x_2.$$

S_2	$c_2x_2 + F_3(s_2 - w_2x_2)$		opt	
	$x_2 = 0$	$x_2 = 1$	$F_2^*(S_2)$	x_2^*
0	$0 + 0 = 0$	–	0	0
1	$0 + 30 = 30$	–	30	0
2	$0 + 60 = 60$	–	60	0
3	$0 + 90 = 90$	$80 + 0 = 80$	90	0
4	$0 + 120 = 120$	$880 + 30 = 110$	120	0
5	$0 + 150 = 150$	$80 + 60 = 140$	150	0

Значение условного оптимума $F_3(S_3)$ берется из предыдущей таблицы.

Этап 3.

$$F_1(S_1) = \max\{c_1x_1 + F_2(S_1 - w_1x_1)\};$$

$$x_1 = 0, 1, 2;$$

$$S_1 = 5.$$

S_1	$c_1x_1 + F_2(S_1 - w_1x_1)$			opt	
	$x_1 = 0$	$x_1 = 1$	$x_1 = 2$	$F_1^*(S_1)$	x_1^*
5	$0 + 150 = 150$	$65 + 90 = 155$	$130 + 30 + 160$	160	2

Определение управляемых переменных начинается с последней таблицы (обратный ход):

$$x_1^* = 2 \rightarrow S_1 = 2 \cdot 2 = 4;$$

$$S_2 = 5 - 4 = 1 \rightarrow x_2^* = 0 \rightarrow S_3 = 1 - 0 = 1 \rightarrow x_3^* = 1.$$

Оптимальное решение $X^* = (2, 0, 1)$.

.....



Контрольные вопросы по главе 6

1. Что из себя представляет динамическое программирование (ДП)?
2. Какова геометрическая интерпретация схемы решения задач ДП?
3. Каковы основные принципы решения задач ДП?
4. Можно ли решать задачи линейного программирования методом ДП?
5. Назовите характерные особенности задач ДП.
6. В чем отличие управляемой переменной от переменной, характеризующей состояние системы в задачах ДП?
7. Запишите уравнение Беллмана для решения задачи о рюкзаке по алгоритму целевой прогонки.
8. Запишите уравнение Беллмана для решения задачи распределения однородного ресурса между подразделениями по алгоритму обратной прогонки.
9. Реализация каких алгоритмов прямой и обратной прогонки для одной и той же задачи приводит к получению различных оптимальных решений?

Глава 7

ЗАДАЧИ УПОРЯДОЧЕНИЯ

7.1 Особенности задач упорядочения

Наиболее актуальным классом задач упорядочения *считаются задачи сетевого планирования и теории расписаний* [7, 19]. В задачах сетевого планирования порядок следования операций известен исходя из технологических соображений (например, при строительстве дома). Он зафиксирован в виде *сетевого графика*, но моменты начала операций и сроки их выполнения заранее могут быть и нефиксированными. Суть состоит в выборе моментов начала операций и сроков их выполнения так, чтобы минимизировать общие затраты по реализации комплекса операций. Методы сетевого планирования дают возможность выявлять потребность в ресурсах при заданных сроках выполнения работ.

На практике довольно часто встречаются ситуации, когда не для всех операций можно заранее установить строгую последовательность их выполнения. Например, при подготовке строительства дома завоз некоторых строительных материалов, на первый взгляд, можно осуществить в произвольном порядке. Однако и в таком порядке может прослеживаться рациональность. Раздел исследования операций, который изучает порядок их следования, называется теорией расписаний.

При решении задач сетевого планирования и теории расписаний могут широко применяться методы теории графов, комбинаторного анализа и другие.

7.2 Сетевое планирование

7.2.1 Построение сетевого графика

Ознакомимся с методами сетевого планирования на примере строительства некоторого объекта. В целом, организация, ведущая строительство, заинтересована в том, чтобы заранее знать более или менее точно срок завершения всех работ, какие работы следует выполнять с наибольшей интенсивностью, а где можно и не

слишком торопиться, сняв с них трудовые ресурсы на напряженные работы. Каким образом можно получить необходимую информацию?

Методы сетевого планирования наиболее применимы при решении подобных вопросов. Все мероприятия, осуществляемые при строительстве объекта, в их взаимосвязи можно представить в виде сетевого графика (сетевой модели), включающего работы, события и характеристики работ или событий.

Сетевой график — это ориентированный граф без контуров, дуги которого имеют одну или несколько числовых характеристик. Дугами изображают *работы*, а вершинами — *события*. Работа — любой трудовой процесс или действие, сопровождающееся затратами времени и ресурсов. Событие — итог того или иного процесса, результат выполнения предшествующих ему работ. События изображаются точками, кружками и т. п. В сетевом графике всегда есть исходное и завершающее события.

Пусть дан список и характеристики работ строительства нулевого цикла некоторого объекта (табл. 7.1).

Таблица 7.1 – Список работ

Наименование работ	Продолжительность работы, дней	Интенсивность использования людских ресурсов, чел./дней
1. Подвоз необходимых материалов к строительной площадке	1	5
2. Подведение электричества	3	5
3. Подведение воды	5	10
4. Строительство опалубки	2	8
5. Закладка бетона	6	10

Сетевой график обладает следующими основными свойствами:

- ни одно событие не может произойти до тех пор, пока не будут закончены все входящие в него работы;
- ни одна работа, выходящая из данного события, не может начаться до тех пор, пока не произойдет данное событие.

Процесс построения сетевого графика выполнения работ является наиболее существенной и трудоемкой частью всего исследования. Приступая к составлению сети, необходимо установить, какие работы должны быть завершены раньше, чем начнется данная, какие могут быть начаты после завершения данной, какие могут выполняться одновременно с данной работой. Все рассуждения и выводы могут быть представлены в виде матрицы A смежности работ, состоящей из нулей и единиц и размерностью $(n \times n)$, где n — число работ.

	1	2	3	4	5	
1						$r_1 = 0$
2	1					$r_2 = 1$
3						$r_3 = 0$
4	1					$r_4 = 1$
5	1	1	1			$r_5 = 3$
	$b_1 = 3$	$b_2 = 1$	$b_3 = 1$	$b_4 = 0$	$b_5 = 0$	

В матрице A элемент:

$$a_{ij} = \begin{cases} 1, & \text{если работа } i \text{ стоит впереди работы } j; \\ 0, & \text{в противном случае.} \end{cases}$$

Сумма элементов в матрице A по каждой строке будет означать, сколько работ стоит впереди рассматриваемой работы в сетевом графике, т. е. впереди работы i стоят работы в количестве $r_i = \sum_{j=1}^4 a_{ij}$. Сумма элементов по каждому столбцу будет означать, сколько работ стоят позади рассматриваемой работы, т. е. позади работы j стоят работы в количестве $b_j = \sum_{i=1}^4 a_{ij}$. Матрица смежности A и оценки r_i и b_j дают информацию для построения графа работ (рис. 7.1).

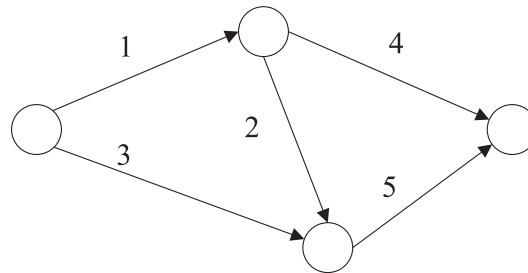


Рис. 7.1 – Граф работ

Цифры на рисунке означают номера работ. Исходное событие начинается работами 1 и 3 (оценки $r_1 = r_3 = 0$). Завершающее событие заканчивается работами 4 и 5 ($b_4 = b_5 = 0$).

Следующим вопросом при построении сетевого графика решается вопрос нумерации событий. Нумерация связана с возможностью применения формализованных процедур расчета сетевого графика. События нумеруются в возрастающем порядке по рангам, начиная с исходного. Чтобы облегчить нумерацию событий в сетях, применяют процедуру разбиения графа на слои. Внутри слоя события могут нумероваться в произвольном порядке, но номера событий предшествующего слоя должны быть меньше номеров последующего. В слой попадают события, для которых нет непосредственных отношений предшествования (дуг). На рисунке 7.2 показан граф, разбитый на слои с нумерацией событий. В каждый слой попало по одному событию.

В дальнейшем обозначения работ будем делать через нумерацию событий. Вместо работы 1 введем обозначение работа (1, 2), вместо работы 3 — работа (1, 3) и т. д.

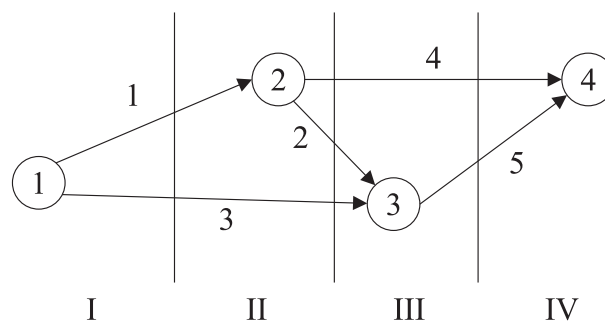


Рис. 7.2 – Разбиение графа на слои

В итоге получим сетевую модель комплекса строительных работ строительства нулевого цикла объекта (рис. 7.3). Над стрелками даны характеристики длительности выполнения работ (дней), а в скобках — численность использования рабочей силы в течение дня.

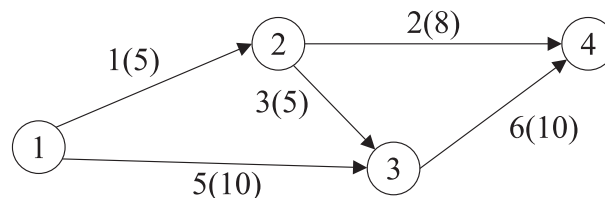


Рис. 7.3 – Сетевой график работ

7.2.2 Расчет параметров сетевого графика

Графический способ

В каждом сетевом графике имеется несколько полных путей (последовательностей работ) от исходного события до конечного, продолжительность каждого из них определяется суммой продолжительностей составляющих их работ. Среди полных путей сетевого графика особое значение имеет наиболее продолжительный из них — **критический путь**. Работы, находящиеся на критическом пути, называют критическими. Критический путь лимитирует выполнение задачи в целом, поэтому любая задержка на критических работах увеличивает время всего процесса. Работы (как и события), не лежащие на критическом пути, имеют резервы времени их выполнения. Поэтому выделяют следующие основные параметры сетевого графика (табл. 7.2).

Таблица 7.2 – Параметры сетевого графика

Событие i	Раннее время $t_p(i)$		Позднее время $t_n(i)$		Резерв $R(i)$
Работа (i, j)	Раннее	Позднее	Раннее	Позднее	Резерв $R(ij)$
	$t_{pn}(ij)$	$t_{пн}(ij)$	$t_{po}(ij)$	$t_{по}(ij)$	
	Начало		Окончание		

$t_p(i)$ — раннее время свершения события i ; $t_n(i)$ — позднее время свершения события i ; $R(i)$ — резерв времени события i ; $t_{pn}(ij)$ — раннее время начала рабо-

ты (i, j) ; $t_{\text{пн}}(ij)$ — позднее время начала работы (i, j) ; $t_{\text{ро}}(ij)$ — раннее время окончания работы (i, j) ; $t_{\text{по}}(ij)$ — позднее время окончания работы (i, j) ; $R(ij)$ — резерв времени работы (i, j) .

Определим параметры для событий и критический путь на графике. На практике получил широкое распространение четырехсекторный способ расчета ранних и поздних сроков свершения событий. При этом способе кружок сетевого графика, обозначающий событие, делится на четыре сектора (рис. 7.4, а). В верхнем ставится номер события i , в левом — наиболее раннее из возможных время свершения события $t_p(i)$, в правом — наиболее позднее из допустимых время свершения события $t_n(i)$, в нижнем — резерв времени данного события $R(i)$.

Раннее время свершения события $t_p(i)$ определяется продолжительностью максимального пути $\max t$ до (i) , предшествующего событию i : $t_p(i) = \max t \text{ до } (i)$.

Послойно, переходя от исходного события до конечного, определим $t_p(i)$. Всегда для начального события $t_p(1) = 0$.

Раннее время свершения события $t_p(i)$ определяется продолжительностью максимального пути $\max t$ до (i) , предшествующего событию i : $t_p(i) = \max t \text{ до } (i)$.

Послойно, переходя от исходного события до конечного, определим $t_p(i)$. Всегда для начального события $t_p(1) = 0$.

Для события 3 (рис. 7.4, б) — $t_p(3) = \max\{(1 + 3), (0 + 5)\} = 5$; для события 4 — $t_p(4) = \max\{(1 + 2), (5 + 6)\} = 11$.

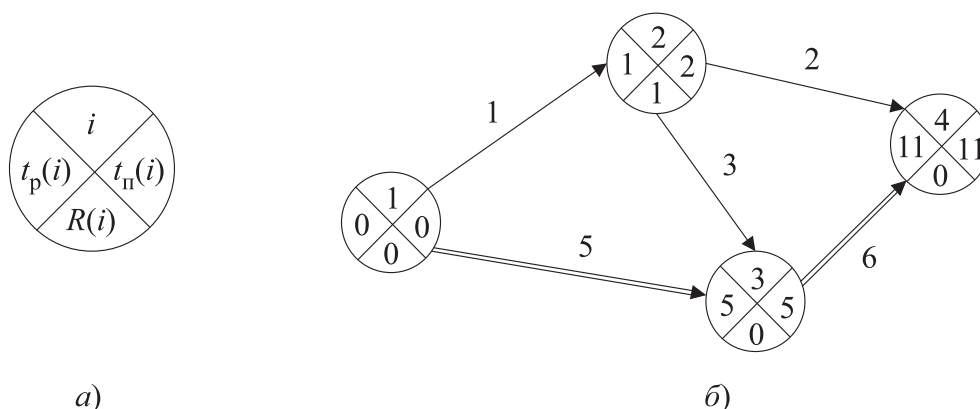


Рис. 7.4 – Графический способ расчета параметров:
а) обозначения в вершине графика; б) сетевой график

Длина критического пути $L_{\text{кр}} = 11$. Послойно, переходя от конечного события до начального, определим $t_n(i)$. Всегда для конечного события $t_n(4) = t(L_{\text{кр}}) = 11$. Позднее время свершения события $t_n(i)$ определяется временем, достаточным для выполнения работ, следующих за этим событием, т. е. зная продолжительность максимального из последующих за событием i путей $\max t$ после (i) и продолжительность критического пути $t(L_{\text{кр}})$, можно найти $t_n(i) = t(L_{\text{кр}}) - \max t \text{ после } (i)$.

Для события 2 — $t_n(3) = 11 - \max\{(3 + 6), 2\} = 2$.

Для критического пути время раннего свершения события $t_p(i)$ равно времени позднего свершения этого события $t_n(i)$, т. е. $t_p(i) = t_n(i)$. Зная ранние и поздние сроки свершения событий сетевого графика, легко выявить резерв времени каждого из них $R(i) = t_n(i) - t_p(i)$.

Резерв времени события показывает максимально допустимое время, на которое можно отодвинуть момент его свершения, не вызывая увеличения критического пути. События критического пути резерва времени не имеют.

Связь параметров сетевого графика для событий и работ показана в таблице 7.3.

Таблица 7.3 – Расчет параметров работ

Время	Начало i	t_{ij}	j Окончание
Раннее 	$t_{рн}(i, j) = t_p(i)$		$t_{ро}(i, j) = t_p(i) + t_{ij}$
Позднее 	$t_{пн}(ij) = t_{п}(j) - t_{ij}$		$t_{по}(ij) = t_{п}(j)$

Резерв времени для работы $R(ij)$ определяется по формуле:

$$R(ij) = t_{по}(ij) - t_{ро}(ij) = t_{п}(j) - t_p(i) - t_{ij}.$$

Табличный способ

При данном способе расчета параметров упорядочение и нумерация вершин графика производится обязательно послойно с выполнением правила: любая последующая вершина имеет больший номер, чем предшествующая, т. е. для любой дуги (i, j) должно выполняться условие $i < j$. Расчеты производятся в таблице $n \times n$, где n — число вершин. Строки и столбцы таблицы соответствуют событиям графика. Клетки главной диагонали таблицы (i, i) назовем главными, а остальные побочными. Для клеток, находящихся выше главной диагонали ($i < j$), номер строки i соответствует номеру начального события, а номер столбца j — номеру конечного. Наоборот, для клеток, расположенных ниже главной диагонали ($i > j$), начальному событию работы соответствует номер столбца j , а конечному — номер строки i .

Расчеты параметров сетевого графика проведем на выше рассмотренном примере (рис. 7.3). Перенесем исходные данные графика в таблицу 7.4.

Таблица 7.4 – Исходные данные

$i \backslash j$	1	2	3	4
1		1	5	—
2	1		3	2
3	5	3		6
4	—	2	6	

Цифры в таблице 7.4 над главной диагональю характеризуют продолжительность выполнения работы (i, j) , под главной диагональю — продолжительность выполнения этой же работы (j, i) , где i — номер строки, j — номер столбца.

Дальнейшее определение параметров в таблице сетевого графика производится в два этапа.

На первом этапе (прямое движение к конечному событию) определяются параметры $t_{po}(i, j)$ и $t_p(j)$. Для конечного события $n - t_p(n) = t_n(n)$.

На втором этапе (обратное движение к начальному событию) определяются параметры $t_{пн}(i, j)$ и $t_n(j)$. Эти параметры будут проставляться (см. табл. 7.5) соответственно выше главной диагонали для $t_{po}(i, j)$, ниже главной диагонали — $t_{пн}(i, j)$ и по главной диагонали — $t_p(i)$, $t_n(i)$, $t_p(j)$, $t_n(j)$. (По аналогии, как проставлены параметры в таблице 7.3.)

Таблица 7.5 – Табличный способ расчета

Обратный ход	Вершины	i	j	Прямой ход
$t_n(1) = t_p(1) = 0$ $t_n(i) = \min_j \{t_{пн}(i, j)\}$ $t_{пн}(i, j) = t_n(i) - t_{ij}$	i	$\frac{t_p(i)}{t_n(i)}$	$\frac{t_{ij}}{t_{po}(i, j)}$	$t_{po}(i, j) = t_p(i) + t_{ij}$
	j	$\frac{t_{ij}}{t_{пн}(i, j)}$	$\frac{t_p(j)}{t_n(j)}$	$t_p(j) = \max_i \{t_{po}(i, j)\}$ $t_p(n) = t_n(n)$

Расчеты параметров сетевого графика представлены в таблицах 7.6–7.7.

Таблица 7.6 – Прямой ход расчета параметров

$i \backslash j$	1	2	3	4
1	0	1	5	—
2	1	1	3	2
3	5	3	5	6
4	—	2	6	11

Таблица 7.7 – Обратный ход расчета параметров

$i \backslash j$	1	2	3	4
1	0	1	5	—
2	1	1	3	2
3	5	3	5	6
4	—	2	6	11

В таблице 7.8 приведены параметры сетевого графика для рассматриваемых работ (V_{ij} — интенсивность потребления ресурсов).

Таблица 7.8 – Значения параметров работ

Работы	t_{ij}	V_{ij}	$t_{рн}$	$t_{ро}$	$t_{пн}$	$t_{по}$	R
1–2	1	5	0	1	1	2	1
1–3	5	10	0	5	0	5	0
2–3	3	5	1	4	2	5	1
2–4	2	8	1	3	9	11	8
3–4	6	10	5	11	5	11	0

7.2.3 Оптимизация распределения трудовых ресурсов

Сетевой график можно применить для определения потребления ресурсов и их возможного распределения. Под оптимальным распределением ресурсов понимается такое размещение работ во времени, которое при заданной интенсивности потребления ресурсов обеспечило бы выполнение всего комплекса работ в минимальный срок.

Сделаем расчет общей интенсивности потребления ресурса по началу и окончанию работ. Для этого построим линейный график (Ганта) комплекса работ (рис. 7.5, а) — сетевой график в календарной шкале времени по ранним срокам начала и окончания работ. На рисунке 7.5, б показана эпюра ресурсов для этого же сетевого графика.

Максимальное использование трудовых ресурсов наблюдается с первого по третий день и равно 23 человекам в день. Если строительная организация располагает численностью меньше 23 человек, то возникает задача перераспределения ресурсов. Пусть ограничение по численности равно 15 человекам. Для того чтобы уложиться в ограничение, необходимо сдвинуть работы (рис. 7.5, а) в пределах возможного таким образом, чтобы суммарная интенсивность была бы не более 15. Большим резервом располагает работа (2, 4). Однако любое ее смещение вправо не позволяет уложиться в ограничение. Но если работу (2, 4), для выполнения которой требуется $2 \cdot 8 = 16$ человеко-дней, растянуть по времени, чтобы интенсивность была не 8 чел./день, а 4 чел./день, т. е. растянуть на $16 : 4 = 4$ дня, и начать ее не раньше $t_{рн} = 4$, то с поставленной задачей мы справимся. На рисунке 7.5, в видно, что было принято решение начать работу (2, 4) с пятого дня. Если бы численность строителей была ограничена еще сильнее, то оптимизацию использования ресурсов пришлось бы производить с изменением критического времени выполнения работ.

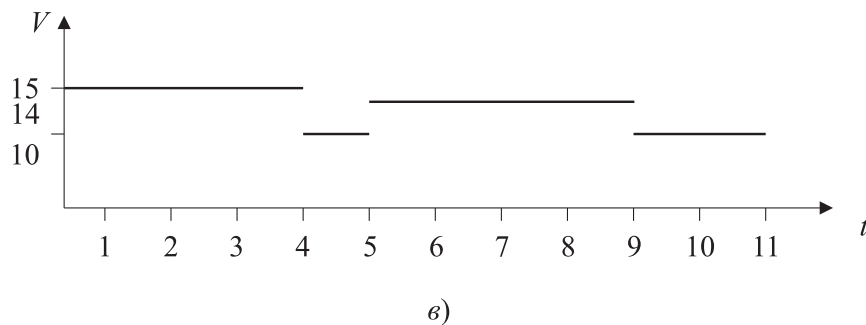
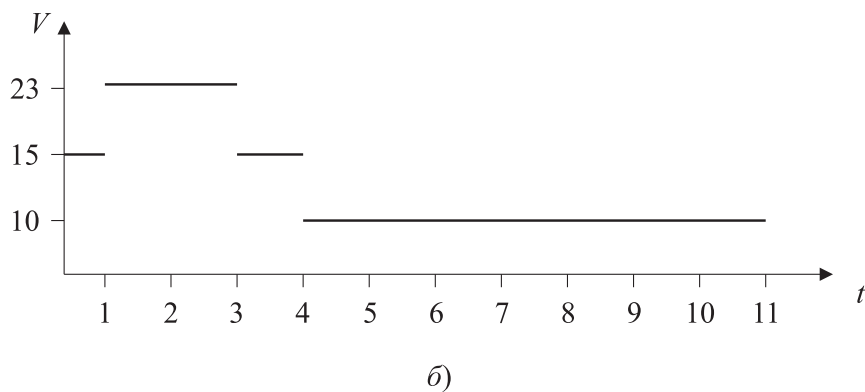
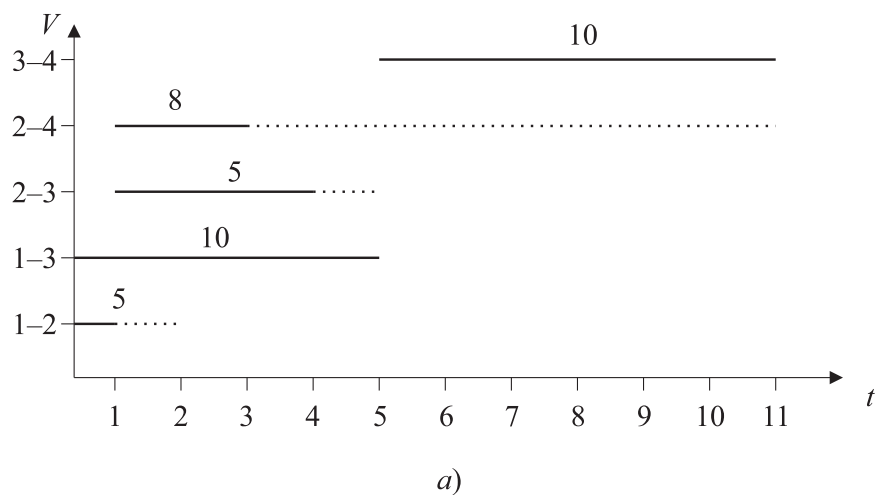


Рис. 7.5 – Графический способ оптимизации использования трудовых ресурсов:
 а) график Ганта; б) эпюра использования ресурсов до оптимизации;
 в) эпюра использования ресурсов после оптимизации

7.3 Составление расписаний

Теория расписаний возникла на базе календарного планирования производства в начале XX в. Ее основоположником считается Гант, впервые предложивший оптимизировать планирование с помощью специальных графиков (Гант-карт).

Задачи теории расписаний связаны с упорядочением операций. Они возникают повсюду, где существует возможность выбора той или иной очередности их выполнения. Под операцией, как и раньше, понимается какое-либо действие, направленное на достижение цели. Множество операций выполняются устройства-

ми (машинами) согласно *расписанию*: для каждой операции на временной оси задается участок, когда операция должна выполняться соответствующей машиной. Расписание обуславливает порядок выполнения операций каждой машиной.

Рассмотрим простейшую задачу теории расписаний: система состоит из одной машины; все операции поступают в систему одновременно, длительности выполнения операций заданы детерминированной величиной. Рассмотрим данную задачу, называемую задачей директора, на конкретном примере [31].

На прием к директору одновременно записались посетители. Секретарь составил список, указав для каждого посетителя ориентировочную продолжительность приема (табл. 7.9), ограничив этот список шестью посетителями, т. к. на прием директору отводилось 2 часа. Секретарю необходимо составить расписание последовательности приема ограниченного числа посетителей, которое приведет к экономии общего времени ожидания посетителей.

Таблица 7.9 – Список посетителей, записавшихся на прием

№ п/п	Фамилия	Продолжительность приема, мин	Время ожидания, мин
1	Антонов	25	0
2	Борисов	15	25
3	Васильев	10	40
4	Гаврилов	5	50
5	Денисов	35	55
6	Егоров	30	90
Суммарное время		120 мин = 2 ч	260 мин = 4 ч 20 мин

При составлении данного расписания секретарь может руководствоваться только длительностью приема посетителей t_i , ($i = \overline{1, 6}$) и порядком (рангом, номером) приема в очереди. Свяжем этот порядок с индексом i . В качестве критерия оптимальности составления такого перестановочного расписания выступает общее время ожидания посетителей в очереди (общее время продолжительности приема посетителей при всех возможных перестановках остается неизменным и равно двум часам).

Сделаем математическую постановку задачи. Пусть i — порядковый номер посещения директора (искомая величина), $1 \leq i \leq 6$. Определим общее время ожидания посетителей Z в очереди как сумму ожидания в очереди каждого из посетителей z_i : $Z = \sum_{i=1}^6 z_i$.

$$z_1 = 0;$$

$$z_2 = t_1;$$

$$z_3 = t_1 + t_2;$$

$$\dots$$

$$z_6 = t_1 + t_2 + t_3 + t_4 + t_5$$

$$\sum_{i=1}^6 z_i = (6-1)t_1 + (6-2)t_2 + (6-3)t_3 + (6-4)t_4 + (6-5)t_5 = \sum_{i=1}^6 (6-i)t_i.$$

Математическая модель примет вид:

$$\min Z = \sum_{i=1}^6 (6-i)t_i \text{ при ограничениях } 1 \leq i \leq 6.$$

Алгоритм решения такой задачи очень прост. Известно, что сумма попарных произведений членов двух числовых последовательностей имеет минимальное значение, если одна из них возрастает, а другая убывает. Так как коэффициенты в целевой $(6-i)$ с увеличением i убывают, то значение Z достигнет минимума в том случае, если значения t_i будут упорядочены по возрастанию, т. е. $t_1 \leq t_2 \leq \dots \leq t_6$. Поэтому расписание, составленное в порядке нарастания продолжительности приема посетителей, даст наименьшее суммарное время ожидания (табл. 7.10).

Таблица 7.10 – Оптимальное расписание приема посетителей

№ п/п	Фамилия	Продолжительность приема, мин	Время ожидания, мин
1	Гаврилов	5	0
2	Васильев	10	5
3	Борисов	15	30
4	Антонов	25	30
5	Егоров	30	55
6	Денисов	35	85
Суммарное время		120 мин = 2 ч	190 мин = 3 ч 10 мин



Контрольные вопросы по главе 7

1. Каковы особенности задач упорядочения?
2. Назовите способы разбиения графа на слои.
3. Назовите основные параметры сетевого графика, укажите их связь и способы определения.
4. Опишите графический способ определения параметров сетевого графика.
5. Опишите табличный способ определения параметров сетевого графика.
6. Что такое график Ганта?
7. Опишите алгоритм оптимизации распределения трудовых ресурсов на графиках Ганта.
8. Сделайте содержательную и математическую постановку задачи директора (приема посетителей).

ЗАКЛЮЧЕНИЕ

В учебном пособии мы рассмотрели вопросы принятия решений в условиях определенности в системах организационного управления при решении основных однокритериальных задач исследования операций, таких как использования ресурсов (планирования производства), задачи транспортного типа, выбора маршрута, упорядочивания и согласования. Для их моделирования и поиска решения были рассмотрены методы математического программирования и теории графов. Так, во второй главе было показано, что задача использования ресурсов может быть представлена в виде задачи линейного программирования (ЗЛП). В этой главе рассмотрены методы и алгоритмы решения этих задач, двойственность ЗЛП, анализ чувствительности оптимального решения к изменению запасов ресурсов и вариациям коэффициентов целевой функции. В третьей главе рассмотрены задачи транспортного типа (транспортная задача, о назначениях, о коммивояжере) в виде ЗЛП, задачи выбора маршрута — в виде задач теории графов. В четвертой главе рассмотрены методы решения целочисленных задач, в пятой — методы условной оптимизации решения задач нелинейного программирования, в шестой — задачи динамического программирования, в седьмой — задачи сетевого планирования и управления.

Для большинства рассмотренных задач, их решения существуют разнообразные программные средства, которыми можно и нужно воспользоваться в той или иной ситуации. В зависимости от проблемы, постановки содержательной и математической задачи принятия решения делается выбор соответствующего программного средства. Для нетиповых задач ПР пишутся оригинальные программные продукты.

ЛИТЕРАТУРА

- [1] Таха Х. Введение в исследование операций : в 2 кн. / Х. Таха. — М. : Мир, 1985. — Кн. 1. — 976 с. ; Кн. 2. — 496 с.
- [2] Акоф Р. Основы исследования операций / Р. Акоф, М. Сасиени. — М. : Мир, 1971. — 536 с.
- [3] Вагнер Г. Основы исследования операций : в 3 т. / Г. Вагнер. — М. : Мир, 1972 — 1973. — Т. 1. — 336 с. ; Т. 2. — 488 с. ; Т. 3. — 503 с.
- [4] Перегудов Ф. И. Введение в системный анализ / Ф. И. Перегудов, Ф. П. Тарасенко. — М. : Высшая школа, 1989. — 367 с.
- [5] Ехлаков Ю. П. Исследование систем управления : конспект лекций / Ю. П. Ехлаков. — Томск : ТУСУР, 1998. — 112 с.
- [6] Оптнер С. Л. Системный анализ для решения деловых и промышленных проблем / С. Л. Оптнер. — М. : Сов. радио, 1969. — 170 с.
- [7] Кремер Н. Ш. Исследование операций в экономике : учеб. пособие для вузов / Н. Ш. Кремер. — М. : ЮНИТИ, 2006. — 402 с.
- [8] Шикин Е. В. Исследование операций : учебник для вузов / Е. В. Шикин. — М. : Проспект, 2006. — 275 с.
- [9] Вилкас Э. И. Решения: история, информация, моделирование. / Э. И. Вилкас, Е. З. Майминас. — М. : Радио и связь, 1981. — 328 с.
- [10] Моргенштерн О. О точности экономико-статистических наблюдений : пер. с англ. / О. Моргенштерн. — М. : Статистика, 1968. — 324 с.
- [11] Кади Дж. Количественные методы в экономике / Дж. Кади. — М. : Прогресс, 1977. — 244 с.
- [12] Фиакко А. Нелинейное программирование / А. Фиакко, Г. Мак-Кормик. — М. : Мир, 1972. — 240 с.
- [13] Кюнц Г. Нелинейное программирование / Г. Кюнц, В. Крелле. — М. : Сов. радио, 1966. — 304 с.
- [14] Форд Л. Поток в сетях / Л. Форд, Д. Фалкерсон. — М. : Мир, 1966. — 277 с.

- [15] Зуховицкий С. И. Математические методы сетевого планирования / С. И. Зуховицкий, И. А. Радчик. — М. : Наука, 1965. — 296 с.
- [16] Ховард Р. А. Динамическое программирование и марковские процессы / Р. А. Ховард. — М. : Сов. радио, 1964. — 189 с.
- [17] Корбут А. А. Дискретное программирование / А. А. Корбут, Ю. Ю. Финкельштейн. — М. : Наука, 1969. — 368 с.
- [18] Ермольев Ю. М. Методы стохастического программирования / Ю. М. Ермольев. — М. : Наука, 1976. — 239 с.
- [19] Танаев В. С. Введение в теорию расписаний / В. С. Танаев, В. В. Шкурба. — М. : Наука, 1975. — 256 с.
- [20] Нейман Дж. Теория игр и экономическое поведение / Дж. Нейман, О. Моргенштерн. — М. : Наука, 1970. — 708 с.
- [21] Оуэн Г. Теория игр / Г. Оуэн. — М. : Мир, 1971. — 230 с.
- [22] Гурин Л. С. Задачи и методы оптимального распределения ресурсов / Л. С. Гурин, Я. С. Дымарский, А. Д. Меркулов. — М. : Сов. радио, 1968. — 463 с.
- [23] Данскин Дж. Теория максимина / Дж. Данскин. — М. : Сов. радио, 1970. — 200 с.
- [24] Гнеденко Б. В. Введение в теорию массового обслуживания / Б. В. Гнеденко, И. А. Коваленко. — М. : Высшая школа, 1987. — 336 с.
- [25] Черненький В. М. Разработка САПР : практическое пособие : в 10 кн. / В. М. Черненький ; под ред. А. В. Петрова. — М. : Высшая школа, 1990. — Кн. 9 : Имитационное моделирование. — 110 с.
- [26] Карлин С. Математические методы в теории игр, программировании и экономике / С. Карлин. — М. : Мир, 1964. — 835 с.
- [27] Авондо-Бодио Дж. Применение в экономике теории графов / Дж. Авондо-Бодио. — М. : Прогресс, 1968. — 162 с.
- [28] Шоломов Л. А. Применение логических методов в задачах последовательного выбора / Л. А. Шоломов. — М. : ВНИИСИ, 1980. — 56 с.
- [29] Статистические и динамические экспертные системы / Попов Э. В. [и др.]. — М. : Финансы и статистика, 1996. — 318 с.
- [30] Трахтенгерц Э. А. Компьютерная поддержка принятия решений / Э. А. Трахтенгерц. — М. : Синтез, 1998. — 376 с.
- [31] Турунтаев Л. П. Оптимизация и математические методы принятия решений : учеб. пособие : в 2 ч. / Л. П. Турунтаев. — Томск : ТМЦДО, 2010. — Ч. 1. — 210 с.
- [32] Банди Б. Основы линейного программирования / Б. Банди. — М. : Радио и связь, 1989. — 176 с.
- [33] Кофман А. Сетевые методы планирования и их применение / А. Кофман, Г. Дебазей. — М. : Мир, 1968. — 182 с.

ГЛОССАРИЙ

Альтернатива — один из вариантов предлагаемых способов решения задачи.

Задача — желаемый результат деятельности, характеризующийся набором количественных данных или параметров этого результата и достижимый за намеченный (заданный) интервал времени в пределах имеющихся ресурсных ограничений.

ЗЛП — задача линейного программирования.

ЗМП — задача математического программирования.

ЗНЛП — задача нелинейного программирования.

ЗЦЛП — задача целочисленного линейного программирования.

Критерий — это правило, по которому отбираются альтернативы, он позволяет оценить альтернативы с точки зрения степени достижения цели.

Исследование операций (ИО) — научная дисциплина, занимающаяся разработкой и практическим применением методов наиболее эффективного управления различными организационными системами.

Линейное программирование — область математики, разрабатывающая теорию и численные методы решения задач нахождения экстремума (максимума или минимума) линейной функции многих переменных при наличии линейных ограничений, т. е. линейных равенств или неравенств, связывающих эти переменные.

ЛПР — лицо, принимающее решение.

Матрица $A = (a_{ij})$ размера $m \times n$ — прямоугольная таблица чисел, содержащая m строк и n столбцов.

n -мерный вектор X — упорядоченная совокупность n действительных чисел $(x_1, x_2, \dots, x_n)$.

Математическая модель Ω — модель, представляющая из себя систему уравнений и ограничений, описывающих состояние исследуемой системы через ее входные и выходные параметры в любой момент времени. При наличии функции предпочтения состояний системы Z (критерий оптимальности, целевая функция) говорят об оптимизационной модели управления (Z, Ω) .

Операция — любое управляемое мероприятие, направленное на достижение цели. Результат операции зависит от способа её проведения, организации, иначе — от выбора некоторых параметров.

Оптимальные решения — решения, которые по тем или иным соображениям предпочтительнее других. Поэтому основной задачей исследования операций является предварительное количественное обоснование оптимальных решений.

План, или допустимое решение задачи линейного программирования, — вектор X пространства E_n , компоненты которого удовлетворяют функциональным и прямым ограничениям задачи.

Принятие решения — действие над множеством *альтернатив* (взаимоисключающих вариантов решений), результатом которого является подмножество отобранных альтернатив, представленное в виде одной альтернативы либо нескольких эффективных несравнимых альтернатив.

Проблема — это потенциальная цель (задача), для которой еще или не найдены альтернативные способы ее достижения, или не представляется возможным выделить ресурсы на альтернативы и проведение операции для ее решения, или то и другое вместе. Проблема всегда связана с несоответствием между действительным и желаемым результатами деятельности.

Проблемная ситуация — совокупность проблемы и ситуации.

Ситуация (условия) — это ограничения, налагаемые социальной и природной средой (внешней средой) и влияющие на исход решения.

Формулы Крамера применяются при решении системы n линейных уравнений с n неизвестными, определитель которой отличен от нуля.

Целевая функция (ЦФ) — это функция, зависящая от управляемых переменных задачи и позволяющая оценить степень достижения цели.

Цель — желаемый результат деятельности, достижимый в пределах некоторого интервала времени.

Экономико-математическая модель — достаточно точное описание исследуемого экономического процесса или объекта с помощью математического аппарата (различного рода функций, уравнений, систем уравнений и неравенств и т. п.).

Эффективность операции — степень её приспособленности к выполнению задачи — количественно выражается в виде критерия эффективности — целевой функции.

Учебное издание

Турунтаев Леонид Петрович

ИССЛЕДОВАНИЕ ОПЕРАЦИЙ

Учебное пособие

Корректор А. Н. Миронова

Компьютерная верстка А. Т. Серикбай

Издано в Томском государственном университете
систем управления и радиоэлектроники.

634050, г. Томск, пр. Ленина, 40

Тел. (3822) 533018.