

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

Кафедра компьютерных систем в управлении и проектировании (КСУП)

В.В. Одинокоев, Н. Ю. Хабибулина

**АВТОМАТИЗИРОВАННЫЕ ИНФОРМАЦИОННО-
УПРАВЛЯЮЩИЕ СИСТЕМЫ**

Учебное пособие

Томск – 2014

В.В. Одинок**ов, Хабибулина Н.Ю.**

Автоматизированные информационно-управляющие системы: учеб. пособие для бакалавров направления подготовки 27.03.04 Управление в технических системах / В.В. Одинок, Н. Ю. Хабибулина. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2014. – 129 с.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1. ПРЕДМЕТ И ЗАДАЧИ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ	5
2. ЭТАПЫ ОПЕРАЦИОННОГО ИССЛЕДОВАНИЯ	7
2.1. Структуризация проблемы.....	7
2.2. Построение математической модели	8
2.3. Примеры задач.....	10
3. ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ	17
3.1 Построение линейных оптимизационных моделей.....	17
3.2 Предварительное преобразование линейной модели	28
3.3. Графическая интерпретация линейных моделей	30
3.4 Симплексный алгоритм	34
3.5. Получение исходного базиса	39
3.6. Анализ моделей на чувствительность и двойственная задача	41
3.7. Примеры задач.....	49
4. СЕТЕВЫЕ ОПТИМИЗАЦИОННЫЕ МОДЕЛИ	55
4.1. Общие свойства сетевых моделей	55
4.2. Модель назначений	56
4.3. Модель выбора кратчайшего пути	60
4.4. Транспортная задача	68
4.5. Задача коммивояжера	84
4.6. Примеры задач.....	89
5. ЦЕЛОЧИСЛЕННОЕ ПРОГРАММИРОВАНИЕ	97
5.1 Общее описание модели.....	97
5.2. Примеры моделей целочисленного программирования	98
5.3 Решение задачи целочисленного программирования	103
5.4 Примеры задач.....	107
5.5. Заключение	111
6. ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ	112
6.1. Общее описание метода	112
6.2 Задача управления запасами	115
6.3 Модель распределения ресурса	122
6.4 Анализ на чувствительность	126
6.5. Заключение	128

ВВЕДЕНИЕ

Данное пособие предназначено для обучения студентов-бакалавров направления подготовки «Управление в технических системах» по одно семестровой дисциплине «Автоматизированные информационно-управляющие системы». Изучение курса заканчивается сдачей экзамена.

Термин «автоматизированное управление» означает совместное участие в выработке управляющих воздействий (решений) человека и ЭВМ. При этом человек выполняет построение математической модели выработки решения, а ЭВМ используется для выполнения вычислений по этой модели. Окончательное утверждение решения остается за человеком.

Полезно сравнить автоматизированное управление с автоматическим управлением, которое предполагает, что окончательное решение выдает сама ЭВМ. Автоматическое решение задач требует наличия гораздо большей определенности в постановке этих задач по сравнению с задачами автоматизированного управления. Этим объясняется использование автоматического управления исключительно в таких технических системах, в которых человеческий фактор отсутствует: человек не только не используется для принятия решения, но и не учитывается как объект управления. Применение автоматического управления не только не исключает, но и делает желательным использование автоматизированного управления, которое используется для получения задающих воздействий или других целевых параметров для автоматических систем управления.

В результате анализа Федерального Государственного образовательного стандарта авторами данного пособия сделан вывод о том, что его тематика очень похожа на тематику известного курса «Исследование операций». Этим объясняется наличие у пособия второго названия, а также то, что это пособие может быть использовано студентами других специальностей как технического, так и экономического профиля, для изучения курса «Исследование операций». Следует отметить, что в данном пособии рассматриваются не все, а только детерминированные методы исследования операций. Вероятностные методы предполагается рассмотреть во второй части пособия.

Несмотря на то, что данное пособие является самодостаточным и содержит всю необходимую информацию для выполнения контрольных работ и для сдачи экзамена, для углубленного знакомства с рассматриваемым предметом рекомендуется знакомство с дополнительной литературой. При этом, так как курс «Исследование операций» сформировался достаточно давно, то можно пользоваться достаточно старыми изданиями. Особо рекомендуется учебник в трех томах [1], в котором оптимальным образом сочетаются доступность изложения и математическая корректность. Книги [2-5], вышедшие в последнее время и в своем большинстве являющиеся переизданиями, также полезны для лучшего знакомства с рассматриваемым предметом.

1. ПРЕДМЕТ И ЗАДАЧИ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ

Краткое определение: *исследованием операций (ИО)* называется научный подход к решению задач организационного управления. Более подробное определение ИО: построение математических моделей, описывающих сложные ситуации, возникающие в разных областях целенаправленной человеческой деятельности и применение этих моделей для обоснованного выбора решений.

Рассмотрим понятие *решение*. Пусть предпринимается какое-то мероприятие (операция), направленное к достижению определенной цели. У лица (или группы лиц), организующего мероприятие, всегда имеется какая-то свобода выбора, т.е. операция является *управляемой*. Это лицо может организовать операцию тем или другим способом, например, распределить имеющиеся средства и выбрать образцы техники, которые будут применены. *Решение* – это и есть какой-то выбор из ряда возможностей, имеющихся у организатора, который, следовательно, является *лицом, принимающим решения (ЛПР)*.

Решения принимаются также давно, как существует само человечество. Например, прежде чем идти охотиться на мамонта, первобытные люди должны были принять решение: в каком месте устроить засаду? Как расставить охотников? Чем их вооружить? Мы с Вами каждый день также принимаем множество решений. Например, как одеться, выходя на улицу, в каком месте перейти улицу и т.д. При этом мы не используем никаких методов ИО, а руководствуемся только своим опытом и здравым смыслом.

Однако бывают решения весьма ответственные. Пусть, например, организуется работа общественного транспорта в новом городе. Необходимо принять ряд решений: по каким маршрутам и какие транспортные средства направить? В каких пунктах сделать остановки? Какова должна быть частота следования машин в зависимости от времени суток и т.д. Эти решения – гораздо сложнее, а главное от них очень многое зависит. Неправильный их выбор может отразиться на жизни целого города. Конечно, и здесь часто опираются лишь на интуицию. Но гораздо разумнее будут решения, если они опираются на математические расчеты. Эти предварительные расчеты позволяют избежать длительного и дорогостоящего поиска решения путем проб и ошибок.

Пусть речь идет о крупномасштабном уникальном мероприятии, например, об отведении части стока северных рек на юг. Здесь недопустимо интуитивное волевое решение. В самом деле, опыт в этом случае молчит, так как мероприятие уникально, а здравый смысл легко может обмануть, если не опирается на расчет. В подобных случаях без использования методов ИО обойтись нельзя, так как ИО и есть своеобразное математическое «примирение» будущих решений, позволяющее избегать серьезных ошибок, на которых учиться просто недопустимо.

Впервые название «Исследование операций» появилось в годы второй мировой войны, когда в вооруженных силах США и Англии были сформированы группы научных работников, которые готовили проекты решений для

командующих боевыми действиями. Эти решения описывали распределение сил и средств по различным объектам. В дальнейшем ИО охватило самые разные области деятельности людей от экономики до охраны природы. Перечислим ряд типичных задач ИО, взятых из разных областей.

1. План снабжения предприятий. Имеется ряд предприятий, потребляющих известные виды сырья, и есть ряд сырьевых баз, которые могут поставлять это сырье предприятиям. Базы связаны с предприятиями какими-то путями сообщения, например, железнодорожными, со своими тарифами. Требуется разработать план снабжения предприятия сырьем: с какой базы, в каком количестве и какое сырье доставлять, чтобы потребности в сырье были обеспечены при минимальных расходах на перевозки.

2. Постройка участка магистрали. Сооружается участок железнодорожной магистрали. В нашем распоряжении определенное количество средств: людей, строительных машин, ремонтных мастерских, грузовых автомобилей и т.д. Требуется спланировать строительство, то есть назначить очередность работ, распределить машины и людей по участкам пути, чтобы оно было завершено в минимальный срок.

3. Продажа сезонных товаров. Для реализации определенной массы сезонных товаров создается сеть временных торговых точек. Требуется выбрать число точек, их размещение, товарные запасы и количество персонала на каждой из них так, чтобы обеспечить максимальную экономическую эффективность распродажи.

4. Снегозащита дорог. Во многих местах метели представляют серьезную помеху движению. Существует ряд способов защиты от снега: профиль дороги, защитные щиты и т.д. Каждый из способов требует известных затрат на сооружение и эксплуатацию. Известны господствующие направления ветров, есть данные о частоте и интенсивности снегопадов. Для каждой из дорог требуется разработать наиболее экономически эффективные средства защиты.

5. Противолодочный рейд. Известно, что в некотором районе находится подводная лодка противника. Группа самолетов получила задание разыскать, обнаружить и уничтожить лодку. Требуется рационально организовать операцию (рейд): выбрать маршруты самолетов, высоту полета, способ атаки так, чтобы с максимальной уверенностью выполнить боевое задание.

6. Выборочный контроль продукции. Завод выпускает определенного вида изделия. Для обеспечения их качества организуется система выборочного контроля. Требуется выбрать размер контрольной партии, набор тестов, правила браковки так, чтобы обеспечить заданный уровень качества при минимальных расходах на контроль.

7. Библиотечное обслуживание. Крупная библиотека обслуживает запросы, поступающие от абонентов. Различные книги пользуются различным спросом. Имеется ряд возможностей распределения книг по стеллажам и хранилищам. Требуется разработать систему библиотечного обслуживания, при которой запросы удовлетворяются максимально быстро.

2. ЭТАПЫ ОПЕРАЦИОННОГО ИССЛЕДОВАНИЯ

2.1. Структуризация проблемы

Всякое операционное исследование, иными словами применение методов ИО, состоит из этапов:

- 1) возникновение проблемы;
- 2) структуризация проблемы (формулировка задачи);
- 3) построение математической модели;
- 4) нахождение математического решения;
- 5) внедрение результатов операционного исследования.

Рассмотрим эти этапы. Строго говоря, этап *возникновение проблемы* принадлежит не операционному исследованию, а предшествует ему. Что такое *проблема*? Проблема, решаемая средствами ИО, заключается всегда в том, что известно желательное поведение (функционирование и развитие) исследуемой системы в будущем, но неизвестно, как это поведение обеспечить, то есть неизвестны управляющие решения, которые мы должны принять сейчас. Итогом выполнения данного этапа является словесное описание желательного поведения исследуемой системы в будущем, из которого совершенно неясно, как это поведение обеспечить.

Целью этапа *структуризации проблемы* является построение структурированной содержательной модели. Термин *содержательная* означает, что для записи модели используются только слова и выражения естественного языка. Структурированная содержательная модель включает:

- а) структуру принимаемого решения;
- б) критерии сравнения отдельных решений;
- в) перечень неуправляемых параметров;
- г) содержательное описание ограничений, которым должно удовлетворять принимаемое решение.

Выявление *структуры решения* означает выявление тех параметров, значения которых может выбирать ЛПР. Данные параметры принято называть *элементами решения* или *управляемыми параметрами (переменными)*. Например, если составляется план перевозок одного типа груза из пунктов отправления A_1, A_2, \dots, A_m в пункты назначения B_1, B_2, \dots, B_n , то элементами решения будут числа x_{ij} , показывающие, какое количество груза будет отправлено из i -го пункта отправления в j -ый пункт назначения. Совокупность чисел $x_{11}, x_{12}, \dots, x_{1n}, \dots, x_{m1}, x_{m2}, \dots, x_{mn}$ образует решение. В дальнейшем будем обозначать любое решение как X .

Критерии сравнения решений W_1, W_2, \dots, W_r выбираются так, чтобы они наилучшим образом описывали желательное поведение исследуемой системы. Прежде, чем выбрать критерии сравнения решений мы должны ответить на вопрос: чего мы хотим и к чему стремимся, предпринимая операцию. Примеры критериев: объем прибыли; количество выпускаемой продукции;

время выполнения операции. Как правило, реальные операции (мероприятия) являются многокритериальными, т.е. принимаемые перед их проведением решения должны сравниваться между собой сразу по нескольким критериям. Например, если в результате операции должно быть спроектировано некоторое техническое устройство, то критериями эффективности решений (проектов устройства) могут являться: затраты на разработку; вес устройства; его мощность и т.д.

Неуправляемые параметры – параметры, значения которых влияют на ход операции, но эти значения не зависят от желания ЛПР. Примерами таких параметров являются: температура наружного воздуха; скорость ветра; количество оборудования на предприятии, если ЛПР не хочет (или не может) развивать основные средства. Обозначим множество неуправляемых параметров как S .

Наличие неуправляемых параметров, как правило, в значительной степени ограничивает множество допустимых решений. Например, нельзя выпускать продукции больше, чем это позволит имеющееся оборудование, или больше, чем ее можно реализовать. Каждое такое ограничение описывается на этапе структуризации проблемы содержательно, т.е. в словесной форме.

2.2. Построение математической модели

На этапе **построения математической модели** необходимо представить критерии сравнения решений количественно, то есть в виде формул. Кроме того, требуется записать в виде формул ограничения, которым должны удовлетворять допустимые решения. Обозначим множество допустимых решений, удовлетворяющих всем ограничениям, как X_d . Реализация данного этапа предполагает обязательный выбор одного или нескольких методов ИО. Эти методы различаются между собой прежде всего типами используемых математических моделей. Все математические модели (методы) ИО можно разделить на классы (рис.1).

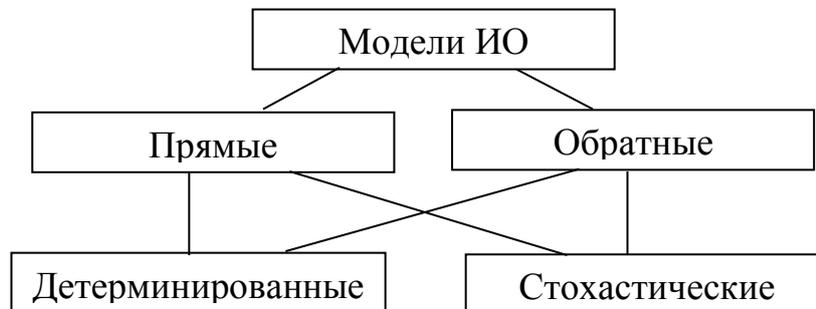


Рис.1. Классификация моделей исследования операций

Прямые методы отвечают на вопрос: что будет, если в заданных условиях мы примем какое-то конкретное решение $X \in X_d$? Иными словами, какие значения примут критерии сравнения решений? Математическая модель в данном случае содержит только зависимости критериев от управляемых и неуправляемых параметров.

Обратные (оптимизационные) методы позволяют ответить на вопрос: как выбрать решение X для того, чтобы некоторый показатель эффективности решений W_L обратился в максимум. (Случай минимизации критерия всегда можно свести к максимизации путем изменения знака W_L). Так как показатель эффективности зависит как от управляемых, так и неуправляемых параметров, то данную модель можно сокращенно записать в виде:

$$\begin{aligned} \text{Max } W_L(S, X) \\ X \in X_d \end{aligned} \quad (2.1)$$

В отличие от прямой модели, обратная модель, во-первых, содержит строго один критерий (целевую функцию) сравнения решений. Это обусловлено тем, что нельзя одновременно максимизировать сразу два критерия, и поэтому все остальные критерии обычно добавляются к ограничениям. Таким образом, множество X_d ограничено не только ранее выявленными ограничениями, но и ограничениями на оценки решения по другим критериям.

В качестве второго признака классификации методов (моделей) ИО возьмем наличие достоверной информации о значениях неуправляемых параметров. Если можно принять допущение о том, что все эти параметры являются детерминированными, то метод относится к классу **детерминированных методов**. При этом параметр считается **детерминированным**, если его значение можно предсказать достаточно точно заранее, то есть при принятии решения. Если хотя бы один неуправляемый параметр не является детерминированным, то соответствующий метод относится к **стохастическим (вероятностным)** методам.

Из четырех классов методов ИО, выделяемых приведенными двумя признаками классификации, прямые методы в условиях определенности слишком просты и поэтому их обычно даже не относят к исследованию операций. Обратные методы в условиях неопределенности распространены мало из-за своей сложности. Наиболее распространены оптимизационные детерминированные методы и прямые стохастические методы.

После того, как математическая модель построена, приступают к **поиску решения модели**. Для этого используется, как правило, ЭВМ. Программа, реализующая метод ИО, или разрабатывается или, что чаще, выбирается готовой. Как правило, в процессе выполнения данного этапа преследуются две цели:

- а) поиск наилучшего решения;
- б) анализ найденного решения на чувствительность к изменениям параметров модели. Данный анализ очень важен, так как многие неуправляемые

параметры модели, которые принимаются детерминированными, на самом деле полностью таковыми не являются.

Что касается *внедрения результатов операционного исследования*, то оно будет проходить тем лучше, чем больше ЛПР привлекался к выполнению всех этапов операционного исследования. Только в этом случае операционное исследование даст ответы именно на те вопросы, которые интересуют ЛПР. Только в этом случае у ЛПР будет доверие к математическому решению.

Начиная со следующего раздела мы приступим к рассмотрению оптимизационных (обратных) детерминированных методов. Вообще говоря, модель (2.1) можно отнести к классу «вариационных задач», хорошо разработанных в математике. Но из-за того, что реальные модели ИО имеют весьма большое число переменных и ограничений, классические методы мало пригодны из-за больших вычислений. Поэтому применяют другие методы, выбираемые в зависимости от типа функции W_L и вида ограничений.

2.3. Примеры задач

Рассматриваемые ниже задачи 1 и 2 представляют собой «игрушечные» примеры применения оптимизационных детерминированных методов, а задачи 3 и 4 – примеры применения прямых стохастических методов.

Задача 2.1

Фирма выпускает три различных продукта 1, 2, 3 из муки. Мука может закупаться у двух различных поставщиков. При этом объемы продуктов 1, 2, 3, которые можно получить из одной тонны муки первого поставщика, отличаются от соответствующих объемов, получаемых из того же количества муки второго поставщика. Соответствующие параметры приведены на рис. 2.

Продукт	Поставщик 1	Поставщик 2	Ограничения на объем продукции
1	0,4	1,2	3,6
2	0,4	0,4	2,4
3	0,6	1,2	4,8
Относительная прибыль	10	20	

Рис. 2. Численные параметры задачи 2.1

Решение

Структурирование проблемы:

- 1) управляемыми переменными (элементами решения) являются:
 - x_1 – количество муки, закупаемого у первого поставщика;
 - x_2 – количество муки, закупаемого у второго поставщика;
- 2) критерием сравнения решений является объем прибыли;

3) существуют три ограничения: объем выпуска каждого продукта не должен превышать величины спроса на этот продукт.

Построение модели:

$$\begin{aligned} 10x_1 + 24x_2 &\Rightarrow \max, \\ 0,4x_1 + 1,2x_2 &\leq 3,6, \\ 0,4x_1 + 0,4x_2 &\leq 2,4, \\ 0,6x_1 + 1,2x_2 &\leq 4,8, \\ x_1 \geq 0, x_2 &\geq 0. \end{aligned}$$

Решение модели. Так как управляемых переменных всего две, решение модели найдем графически (рис.3). Для этого, во-первых, изобразим на плоскости две координатные оси: x_1 и x_2 . Во-вторых, построим на этой координатной плоскости область допустимых решений. Для построения такой области следует построить на плоскости прямые, имеющие уравнения:

$$\begin{aligned} 0,4x_1 + 1,2x_2 &= 3,6, \\ 0,4x_1 + 0,4x_2 &= 2,4, \\ 0,6x_1 + 1,2x_2 &= 4,8, \\ x_1 = 0, x_2 &= 0. \end{aligned}$$

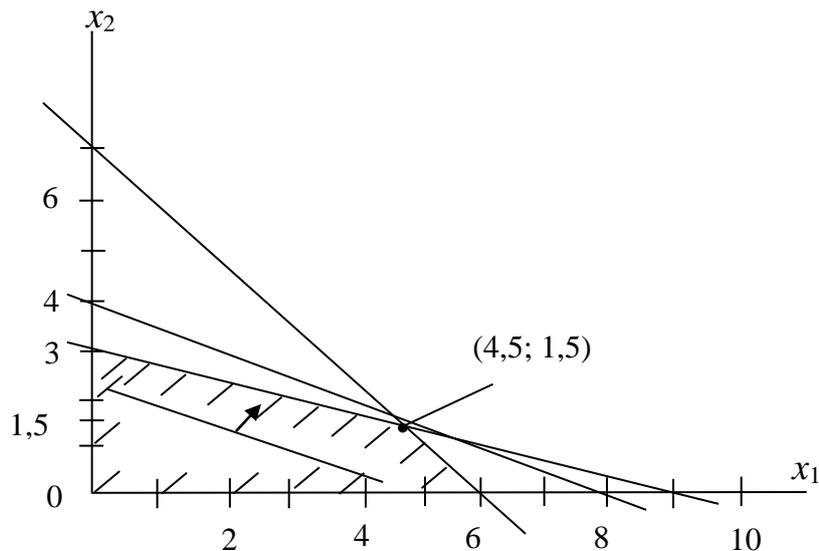


Рис. 3. Графическое решение задачи 2.1

Каждая из этих прямых делит координатную плоскость на две полуплоскости, на одной из которых соответствующее ограничение модели выполняется, а на другой – нет. Для определения полуплоскости, на которой ограничение выполняется, достаточно взять любую точку, например $(0,0)$, и проверить, выполняется ли соответствующее неравенство для координат этой точки. На рис. 3 область допустимых решений показана с помощью штриховых линий.

В-третьих, для поиска в области допустимых решений наилучшей точки (точек), которой соответствует наибольшее значение целевой функции, необходимо эту целевую функцию построить графически. Это можно сделать, ес-

ли присвоить целевой функции какое-то значение, например, 40. В этом случае получим уравнение прямой, называемой *линией равного уровня целевой функции*:

$$10x_1 + 20x_2 = 40.$$

Построив на плоскости эту прямую, а затем линии равного уровня для других значений целевой функции, нетрудно убедиться, что все эти прямые параллельны друг другу. Поэтому для определения оптимальной точки достаточно построить любую линию равного уровня целевой функции, а затем мысленно двигать эту линию параллельно самой себе в сторону увеличения целевой функции (на рис.3 это направление показано стрелкой). Нетрудно убедиться, что такое движение может продолжаться до достижения точки (4,5; 1,5). В данной точке целевая функция имеет наибольшее значение: $10 \times 4,5 + 20 \times 1,5 = 75$.

Задача 2.2

Предприятие потребляет некоторое сырье в объеме M единиц в месяц. Норма потребления не меняется со временем. Время доставки сырья по заказу точно известно. Предприятие может заказывать сырье в любом количестве. При этом излишки могут храниться на складе.

Известно, что каждый очередной заказ сопряжен с некоторыми вполне определенными накладными расходами в размере K . При этом под *накладными расходами* понимаются затраты на оформление заказа и его транспортировку, если она не зависит от объема заказываемой партии. Цена единицы сырья равна c . За хранение единицы сырья в течение месяца взимается плата h .

Решение

Структурирование проблемы:

1) структура решения (управляемые переменные):

x – объем заказываемой партии сырья;

2) критерий оценки вариантов x :

W – среднемесячные затраты, связанные с обеспечением сырьем;

3) неуправляемый параметр: объем сырья, потребляемый в месяц.

Построение модели. Среднемесячные затраты на приобретение сырья включают: а) накладные расходы; б) оплата за само сырье; в) расходы на хранение сырья. Нетрудно заметить, что для сокращения ежемесячных накладных расходов выгодно увеличить объем заказываемой партии сырья, а для сокращения затрат на хранение, наоборот, объем этой партии следует уменьшать. Прежде, чем записать количественные зависимости перечисленных элементов затрат от x , рассмотрим график изменения запаса сырья во времени (рис. 4).

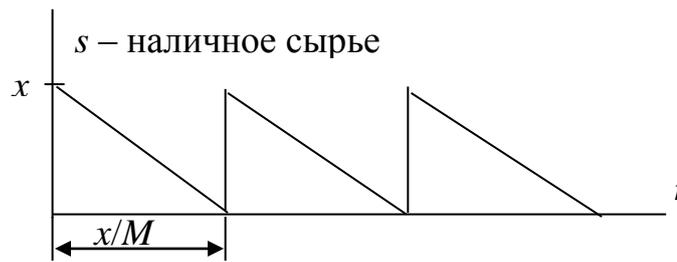


Рис. 4. График изменения запаса сырья во времени

Из этого графика видно, что одной партии сырья хватает на x/M недель. При этом средимесячные накладные расходы: $K/(x/M) = K \cdot M/x$. Кроме того нетрудно заметить, что среднее количество сырья на складе — $x/2$. Средние затраты на его хранение — $h \cdot x/2$. Так как среднимесячные расходы на покупку сырья — $c \cdot M$, то общие ежемесячные затраты будут:

$$W = \frac{KM}{x} + cM + \frac{h \cdot x}{2} \Rightarrow \min.$$

Решение модели. Так как модель не имеет ограничений, для поиска оптимального значения x будем использовать результат, известный из классической математики: необходимым условием оптимума функции W является равенство нулю ее первой производной. Поэтому находим $\frac{dw}{dx}$ и приравниваем

0. Отсюда: $x = \sqrt{\frac{2KM}{h}}$.

Задача 2.3

ЛПР намерен подыскать себе нового бухгалтера путем собеседования с кандидатурами на эту должность. По своему опыту ЛПР знает, что с вероятностью 0,1 кандидатура обладает отличными данными, с вероятностью 0,5 — хорошими данными, а с вероятностью 0,4 — посредственными данными. Эти категории секретарей ЛПР оценивает по своей относительной шкале соответственно в 3, 2 и 1 балл.

ЛПР намерен рассмотреть не более трех кандидатур. К сожалению, если девушка не принимается на работу, она тут же устраивается на другую работу. Следовательно, решение должно приниматься немедленно.

Если на первом или втором собеседовании получена кандидатура с отличными данными, то она, естественно, принимается. Если на таком собеседовании кандидатура оценена в 1 балл, то она отвергается. Проблема состоит в том, как поступить на первом (втором) собеседовании, если получена кандидатура с хорошими данными. Если такая кандидатура будет отвергнута, то возможно придется довольствоваться посредственной кандидатурой, так как на третьем собеседовании любая кандидатура должна быть принята.

Решение

Структурирование проблемы:

- 1) структура решения (управляемые переменные):
 - x_1 – выбор на 1-м собеседовании при получении хорошей кандидатуры (1 – принять, 0 – отвергнуть);
 - x_2 – выбор на 2-м собеседовании при получении хорошей кандидатуры (1 – принять, 0 – отвергнуть);
- 2) критерий оценки вариантов x :
 - W – ожидаемая оценка (в баллах) принятой кандидатуры;
- 3) ограничения: $x_1 = 0; 1$ $x_2 = 0; 1$.

Построение модели. Будем использовать не математическую, а графическую модель, в которой множество допустимых решений представлено в виде дерева альтернатив (рис.5). При этом кружки обозначают моменты принятия решения, то есть собеседования. Дуги, выходящие из кружка, обозначают возможные исходы собеседования. Число на дуге есть вероятность соответствующего исхода. Квадратик на дуге означает момент принятия решения в случае соответствующего исхода. При этом, если из квадратика выходят более одной дуги, то они соответствуют возможным вариантам решения. Если из квадратика выходит всего одна дуга, то никакого выбора в данный момент нет. Точка означает завершение всей операции. На числа, записанные около кружков и в квадратиках, пока не обращайте внимание.

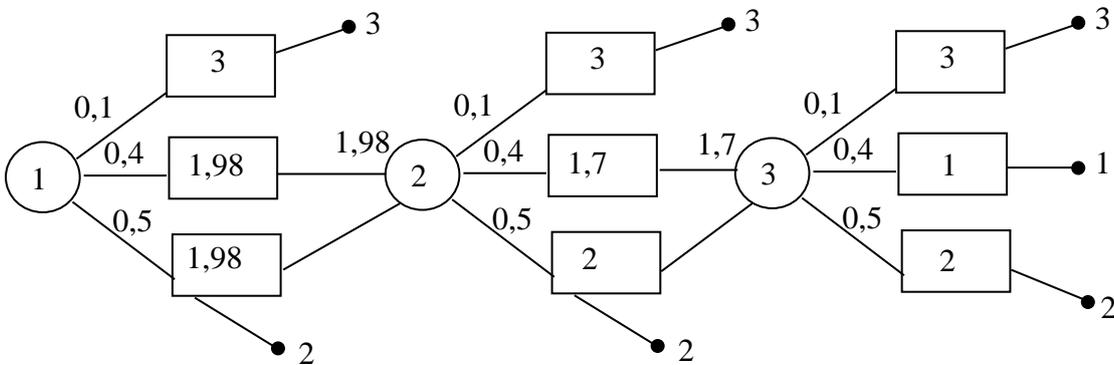


Рис. 5. Дерево альтернатив для задачи выбора бухгалтера

Решение модели. Допустим, что уже рассматривается третья кандидатура. Найдем ожидаемую оценку по формуле расчета математического ожидания этой оценки, то есть как сумму произведений вероятностей исходов на величины этих исходов: $3 \times 0,1 + 2 \times 0,5 + 1 \times 0,4 = 1,7$. Проставим 1,7 около узла 3 (рис. 5). Следовательно, если при втором собеседовании выпало 2, то следует прекращать собеседование ($x_2 = 1$).

Теперь мы можем найти ожидаемый доход от двух (второго и третьего) собеседований: $3 \times 0,1 + 2 \times 0,5 + 1,7 \times 0,4 = 1,98$. Это число проставим около кружка 2.

Рассмотрим самое первое собеседование. Если выпала оценка 2, то собеседование следует продолжить ($x_1=0$). При этом ожидаемая оценка секретаря, выявленного в результате всех трех собеседований:

$$3 \times 0,1 + 1,98 \times 0,9 = 0,102.$$

Выполним простейший анализ чувствительности решения, ответив на вопрос: как влияет число собеседований на ожидаемый результат? Разность $2,082 - 1,98 = 0,102$ есть выгода от третьего собеседования. Аналогично выгода от второго собеседования: $1,98 - 1,7 = 0,28$. Таким образом, выгода от каждого следующего собеседования падает.

Задача 2.4

Кондитерская фирма открывает пункт предварительных заказов на изготовление кондитерских изделий. ЛПР хочет определить число линий телефонной связи (одна или две), необходимое для обслуживания будущих клиентов. Допустим, что известны статистические распределения интервалов времени: 1) между двумя последовательными телефонными звонками; 2) между моментами начала и завершения телефонного разговора.

Решение

Структурирование проблемы:

- 1) структура решения (управляемые переменные):
 x – число линий связи;
- 2) критерии оценки вариантов x :
 W_1 – относительная доля не обслуженных телефонных звонков;
 W_2 – доля времени, когда все линии заняты;
- 3) неуправляемые параметры: интервалы времени между телефонными звонками и продолжительности телефонных разговоров;
- 4) ограничение: $x=1; 2$.

Построение модели. Будем использовать самый распространенный прямой стохастический метод - метод имитационного моделирования. ***Имитационная модель*** – алгоритм, преобразующий значения управляемых переменных и неуправляемых параметров в численные значения критериев W_k .

Причем для каждого набора численных значений управляемых переменных, который интересует ЛПР, требуется осуществить свое выполнение имитационной модели. В результате такого выполнения выявляются численные оценки решения по критериям W_k . После этого выбор лучшего решения выполняет сам ЛПР.

Обычно один или несколько неуправляемых параметров имеют стохастический характер, и поэтому для получения оценок решения требуется выполнить многократную генерацию таких параметров. Естественно, что генерация неуправляемого стохастического параметра должна выполняться в соответствии с законом его статистического распределения. Такая генерация выполняется или стандартной подпрограммой, или подпрограммой, разрабатываемой специально для данной имитационной модели.

Допустим, что в рассматриваемой задаче статистические распределения обоих неуправляемых параметров одинаковы (рис. 6). Для наглядности предположим, что параметры генерируются не программно, а физически. Для этого поместим в центр кружка, соответствующего статистическому распределению, небольшую стрелку, которую можно вращать вручную. Сектор кружка, на который укажет стрелка после своей остановки, и есть очередное значение генерируемого случайного параметра.

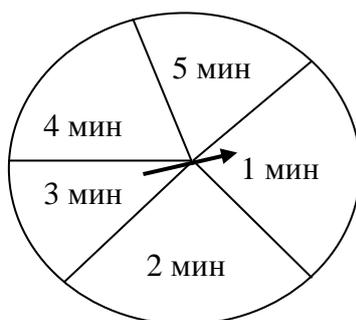


Рис. 6. Статистическое распределение интервалов времени

Решение модели. Выполним имитацию работы первого варианта системы (с одной линией связи) и второго варианта (с двумя линиями связи). Для этого выполним имитацию последовательности телефонных звонков путем осуществления серии последовательных резких вращений стрелки внутри диаграммы (рис.6). Результаты этой имитации приведены на рис. 7. Период имитации – 16 минут, за это время пришло 10 звонков.

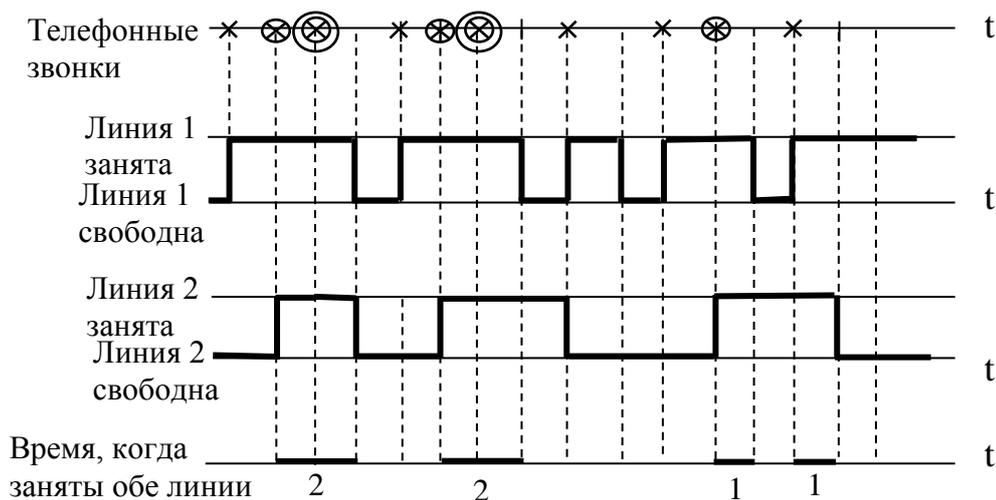


Рис. 7. Имитация прихода телефонных звонков и их обработки

На рис. 7 приведены также результаты обслуживания клиентов одной и двумя линиями связи. При наличии одной линии ее переключение в состоя-

ние «занято» происходит в том случае, если в момент прихода очередного звонка линия была свободна. Для имитации продолжительности каждого телефонного разговора опять вращается стрелка в центре диаграммы. Имитация работы второй линии связи производится аналогично. Единственное отличие: переключение второй линии в состояние «занято» выполняется лишь тогда, когда первая линия занята. На рис. 7 также показано время, когда заняты обе линии связи.

Выполнив имитацию работы обеих линий связи, нетрудно подсчитать значения критериев для обоих вариантов системы:

$$1) \text{ доля не обслуженных звонков: } W_1 = \frac{5}{10} = \frac{1}{2} = 0,5 (50\%) ,$$

$$\text{доля времени, когда линия занята: } W_2 = \frac{12}{16} = 0,75 (75\%) ;$$

$$2) \text{ доля не обслуженных звонков: } W_1 = \frac{3}{10} \approx 0,33 (33\%) ,$$

$$\text{доля времени, когда линия занята: } W_2 = \frac{6}{16} \approx 0,375 (38\%) .$$

Обратим внимание, что по критерию W_1 лучше вариант 2, а по критерию W_2 – вариант 1. Как и любой прямой метод, имитационное моделирование не дает ответ на вопрос о том, какой вариант лучше. Ответ на этот вопрос должен дать сам ЛПР, используя полученные при моделировании оценки критериев.

3. ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

3.1 Построение линейных оптимизационных моделей

Линейной оптимизационной моделью называется модель, целевая функция которой и все ограничения линейны относительно управляемых переменных. Это значит, что, во-первых, все управляемые переменные присутствуют в модели в первой степени, а во-вторых, нет произведений управляемых переменных. Раздел ИО, в котором рассматриваются вопросы построения и исследования линейных оптимизационных моделей, называется *линейным программированием*. При этом термин «программирование» используется как синоним слова «планирование». Рассмотрим несколько типичных задач организационного управления, для решения которых могут быть полезны линейные оптимизационные модели.

3.1.1. Задачи распределения ресурсов

Общая постановка задачи. Наиболее типичная постановка задачи распределения ресурсов следующая. На предприятии, выпускающем неоднородную продукцию, ЛПР стремится определить, какими должны быть интенсив-

ности производства для каждого продукта, или, если плановый период конечен, объемы выпуска каждой продукции за плановый период. Эти интенсивности (объемы) ограничены технологическими и другими внутренними для данного предприятия условиями, заданными в виде линейных соотношений (равенств или неравенств). В рамках этих ограничений ЛПР пытается оптимизировать некоторую конкретную целевую функцию.

В качестве конкретного примера общей задачи распределения ресурсов рассмотрим следующий. Пусть предприятие имеет возможность организовать от одного до четырех различных типов производственно-технологических процессов. Технологические процессы первого и второго типов ориентированы на получение продукции A , а технологические процессы третьего и четвертого типов – на получение продукции B . Организация каждого технологического процесса связана с трудозатратами (измеряемыми в человеко-неделях), количеством (в единицах веса) потребляемого в течение недели материала Y и количеством (в ящиках) потребляемого в течение недели материала Z . При составлении производственного плана на неделю диапазон возможностей ЛПР ограничен как за счет людских ресурсов, так и за счет потребляемых видов сырья. Производственно-экономические показатели и все имеющиеся ограничения представлены на рис. 8.

	На единицу продукции A		На единицу продукции B		Имеется в наличии
	Технолог. процесс 1	Технолог. процесс 2	Технолог. процесс 3	Технолог. процесс 4	
Количество человеко-недель	2	2	2	2	≤ 20
Количество материала Y	5	3	1	1	≤ 100
Количество материала Z	5	5	10	20	≤ 80
Доход с единицы продукции	3	4	7	9	Максимизировать
Объем выпуска продукции	x_1	x_2	x_3	x_4	

Рис. 8. Параметры примера распределения ресурсов

Используя данные на рис. 8, можно было бы сразу же записать линейную модель. Но прежде, чем это сделать, мы рассмотрим два допущения, принимаемые при построении линейных моделей – аксиомы линейности, играющие исключительно важную роль. Это аксиома делимости и аксиома аддитивности.

Делимость: для каждого производственно-технологического процесса суммарное количество каждого из потребляемых ресурсов и соответствующая прибыль строго пропорциональны объему выпускаемой продукции. То есть все показатели производственно-технологического процесса могут быть увеличены или уменьшены при сохранении их взаимной пропорциональности. Например, если увеличить вдвое объем выпускаемой продукции, то вдвое увеличится потребление всех ресурсов, вдвое возрастет и прибыль. В частности, если $x_1=10$, необходимо затратить 20 человеко-недель, 50 кг материала Y и 50 ящиков материала Z . Если $x_1=20$, затраты составят соответственно 40, 100 и 100. Условие делимости предполагает также, что все переменные могут принимать не только целые, но и дробные значения. Например, допустимы $x_2=2,5$ и $x_4=10/3$.

Аддитивность: если значение каждой из управляемых переменных x_j определено (то есть определен соответствующий объем выпуска за весь плановый период или интенсивность выпуска), то полное количество каждого из потребляемых ресурсов равно сумме одноименных ресурсов, затраченных при реализации всех применявшихся технологических процессов. А полная прибыль равна сумме прибылей от отдельных технологических процессов. Например, для производства единицы продукции A в ходе технологического процесса 1 и единицы продукции B в ходе технологического процесса 3 требуется 4 человеко-недели, 6 кг материала Y и 15 ящиков материала Z . При этом доход составляет 10 рублей.

Принятие допущений о делимости и аддитивности эквивалентно утверждению о том, что соответствующая математическая модель является линейной. Если данные аксиомы для решаемой проблемы слишком грубы, то приходится применять более сложные математические модели, в которых выполнение данных аксиом не требуется. Так как для нашего примера делимость и аддитивность выполняются, соответствующая математическая модель линейна:

$$\begin{aligned} 3x_1 + 4x_2 + 7x_3 + 9x_4 &\Rightarrow \max, \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 &\leq 20, \\ 5x_1 + 3x_2 + x_3 + x_4 &\leq 100, \\ 5x_1 + 5x_2 + 10x_3 + 20x_4 &\leq 80, \\ x_j &\geq 0, \quad j=1,2,3,4. \end{aligned}$$

Задача о загрузке оборудования. Ткацкая фабрика располагает двумя видами станков, из них N_1 станков типа 1 и N_2 станков типа 2. Станки могут производить три вида тканей: T_1 , T_2 , T_3 , но с разной производительностью. Данные a_{ij} производительности станков приведены на рис. 9. При этом i – тип станка, j – тип ткани. Каждый метр ткани T_1 приносит фабрике доход c_1 , вида T_2 – доход c_2 , T_3 – доход c_3 .

Тип станка	Вид ткани		
	T_1	T_2	T_3
1	a_{11}	a_{12}	a_{13}
2	a_{21}	a_{22}	a_{23}

Рис. 9. Параметры задачи загрузки оборудования

Фабрике предписан план, согласно которому она должна производить в месяц не менее b_1 метров ткани T_1 , b_2 метров ткани T_2 , b_3 метров ткани T_3 . Количество метров каждого вида ткани не должно превышать соответственно d_1, d_2, d_3 метров. Кроме того, все без исключения станки должны быть загружены. Требуется так распределить загрузку станков производством тканей T_1, T_2, T_3 , чтобы суммарный месячный доход был максимален.

На первый взгляд в качестве управляемых переменных можно взять подобно предыдущей задаче объемы выпускаемой продукции (т.е. тканей). Но это неправильно. Для того чтобы объем выпуска какой-то продукции можно было взять в качестве элемента решения, необходимо убедиться, что при производстве этого вида продукции не используются хотя бы два взаимозаменяемых ресурса. Если это так, то в качестве управляемых переменных должны быть взяты объемы этих взаимозаменяемых ресурсов, идущих на производство данного вида продукции.

Рассмотрим, например, выпуск ткани T_1 . Он может осуществляться с помощью затрат двух взаимозаменяемых ресурсов – станков 1 и 2. Поэтому следует взять две управляемые переменные: x_{11} – количество станков 1, занятых изготовлением ткани T_1 , x_{21} – количество станков 2, занятых изготовлением ткани T_1 . Аналогично рассмотрев остальные станки получаем 6 управляемых переменных: $x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}$.

Приступим к записи математической модели. Так как критерием сравнения решений является прибыль, запишем математическую зависимость этого показателя от управляемых переменных. Найдем сначала общее количество ткани T_1 , произведенное всеми станками, оно равно $a_{11}x_{11} + a_{21}x_{21}$ принося фабрике доход $c_1(a_{11}x_{11} + a_{21}x_{21})$. Общий доход фабрики будет:

$$W = c_1(a_{11}x_{11} + a_{21}x_{21}) + c_2(a_{12}x_{12} + a_{22}x_{22}) + c_3(a_{13}x_{13} + a_{23}x_{23}) \Rightarrow \max$$

Сокращенно целевая функция имеет вид:

$$\sum_{j=1}^3 c_j \sum_{i=1}^2 a_{ij} x_{ij} \Rightarrow \max. \quad (3.1)$$

Запишем ограничения, обеспечивающие выполнение плана:

$$\begin{aligned}
 a_{11}x_{11} + a_{21}x_{21} &\geq b_1, \\
 a_{12}x_{12} + a_{22}x_{22} &\geq b_2, \\
 a_{13}x_{13} + a_{23}x_{23} &\geq b_3.
 \end{aligned}
 \tag{3.2}$$

Обратим внимание, что каждая переменная модели записана в отдельном столбце. Это позволяет лучше воспринимать структуру модели, и мы будем придерживаться такой записи и в дальнейшем. Ограничения, описывающие перевыполнение плана:

$$\begin{aligned}
 a_{11}x_{11} + a_{21}x_{21} &\leq d_1, \\
 a_{12}x_{12} + a_{22}x_{22} &\leq d_2, \\
 a_{13}x_{13} + a_{23}x_{23} &\leq d_3.
 \end{aligned}
 \tag{3.3}$$

Ограничения, обеспечивающие полную загрузку оборудования:

$$\begin{aligned}
 x_{11} + x_{12} + x_{13} &= N_1, \\
 x_{21} + x_{22} + x_{23} &= N_2.
 \end{aligned}
 \tag{3.4}$$

Записав условие неотрицательности управляемых переменных:

$$x_{ij} \geq 0, \quad i = 1, 2, \quad j = 1, 2, 3, \tag{3.5}$$

получили модель линейного программирования (3.1)-(3.5).

3.1.2. Задачи получения смесей

Задача рационального составления комбикорма. Пусть крупная свиноферма имеет возможность покупать от одного до трех различных видов зерна и готовить различного вида смеси (комбикорма). Различные зерновые культуры содержат разное количество питательных компонентов. Существенными являются четыре компонента. Управляющему фермой известно минимальное содержание каждого компонента, которое должен содержать комбикорм. Известно, что период планирования равен двум неделям, то есть зерно покупается в количестве, достаточном для прокорма свиней в течение двух недель. Все данные, относящиеся к рассматриваемой задаче, приведены на рис. 10.

	Одна тонна			Минимальные суммарные потребности на планируемый период
	Зерна 1	Зерна 2	Зерна 3	
Компонент А	4	6	14	2500
Компонент В	2	2	0	500
Компонент С	10	6	0	1800

Компонент D	1,2	0,5	2	465
Затраты на одну тонну в рублях	82	70	192	минимизировать
Количество в т.	x_1	x_2	x_3	

Рис. 10. Данные задачи получения комбикорма

Приняв допущение, что выполняются аксиомы делимости и аддитивности, сформулируем линейную оптимизационную модель:

$$\begin{aligned}
 82x_1 + 70x_2 + 192x_3 &\rightarrow \min, \\
 4x_1 + 6x_2 + 14x_3 &\geq 2500, \\
 2x_1 + 2x_2 &\geq 500, \\
 10x_1 + 6x_2 &\geq 1800, \\
 1,2x_1 + 0,5x_2 + 2x_3 &\geq 465, \\
 x_j &\geq 0, \quad j=1,2,3.
 \end{aligned}$$

Заметим, что данная модель имеет 4 ограничения и всего 3 управляемых переменных. В действительности обычно бывает наоборот: число переменных превышает число ограничений.

Нетрудно заметить, что одним из допустимых решений является: $x_1=625$, $x_2=x_3=0$. Оптимальное решение оказывается следующее: $x_1=200$, $x_2=50$, $x_3=100$.

Задача составления жидких смесей. Еще один класс линейных моделей возникает при решении проблемы, связанной с приготовлением смеси различных жидкостей с целью получения пользующейся спросом продукции. Примеры: 1) смесь нескольких сортов нефти; 2) смесь доведенных до расплавленного состояния различных металлов.

Пусть некоторое предприятие изготавливает три типа химических продуктов, каждый из которых является смесью двух компонентов. Сформулируем данную задачу в общем виде, т.е. используя символьные обозначения.

В качестве управляемой переменной x_{ij} ($i=1,2$, $j=1,2,3$) возьмем объем (в литрах) i -го химического компонента, используемого для получения j -го продукта.

Первая пара ограничений относится к объемам потребляемых химических компонентов:

$$\begin{aligned}
 x_{11} + x_{12} + x_{13} &\leq S_1, \\
 x_{21} + x_{22} + x_{23} &\leq S_2,
 \end{aligned}$$

где S_i – объем i -го химического компонента, которым располагает предприятие в начале планового периода. Первое из ограничений означает, что суммарное количество химического компонента 1, используемого для получения продуктов 1,2,3 не может превышать S_1 .

Вторая группа ограничений отражает требование, чтобы выпуск каждого вида продукта хотя бы в минимальной степени удовлетворял бы спрос:

$$\begin{aligned} x_{11} + x_{21} &\geq D_1, \\ x_{12} + x_{22} &\geq D_2, \\ x_{13} + x_{23} &\geq D_3, \end{aligned}$$

где D_j – минимальные потребности покупателей в j -м продукте в течение планового периода. Первое из ограничений означает, что суммарное количество продукта 1, получаемого путем смешивания химических компонентов 1 и 2, должно удовлетворять по крайней мере минимальный спрос D_1 .

Следующие ограничения описывают требования к содержанию в производимых продуктах каких-то веществ. Первое из них требует, чтобы относительное содержание в продукте 1 некоторого вещества A было не меньше, чем r_1 :

$$\frac{a_1 x_{11} + a_2 x_{21}}{x_{11} + x_{21}} \geq r_1,$$

где a_i – количество вещества A в одном литре компонента i . Другое ограничение отражает требование, чтобы содержание в продукте 2 вещества B не превосходило r_2 :

$$\frac{b_1 x_{12} + b_2 x_{22}}{x_{12} + x_{22}} \leq r_2,$$

то есть имеет ограничение сверху. Нетрудно убедиться, что последние два ограничения можно привести к обычному линейному виду:

$$\begin{aligned} (a_1 - r_1)x_{11} + (a_2 - r_1)x_{21} &\geq 0, \\ (b_1 - r_2)x_{12} + (b_2 - r_2)x_{22} &\leq 0. \end{aligned}$$

Наконец, последнее ограничение может отражать требование, чтобы отношение между объемами компонентов при изготовлении продукта 3 не было меньше, чем r_3 :

$$\frac{x_{13}}{x_{23}} \geq r_3, \quad \text{или} \quad x_{13} - r_3 x_{23} \geq 0.$$

Обозначим через p_{ij} доход с единицы компонента i , затраченной на изготовление продукции j , тогда рассматриваемую модель можно представить в виде:

$$\begin{aligned}
 & p_{11}x_{11} + p_{12}x_{12} + p_{13}x_{13} + p_{21}x_{21} + p_{22}x_{22} + p_{23}x_{23} \Rightarrow \max, \\
 & \quad 1x_{11} \quad + 1x_{12} \quad + 1x_{13} \quad \leq S_1, \\
 & \quad \quad \quad \quad \quad \quad \quad 1x_{21} \quad + 1x_{22} \quad + 1x_{23} \leq S_2, \\
 & \quad 1x_{11} \quad \quad \quad \quad \quad + 1x_{21} \quad \quad \quad \quad \geq D_1, \\
 & \quad \quad \quad 1x_{12} \quad \quad \quad \quad \quad + 1x_{22} \quad \quad \quad \geq D_2, \\
 & \quad \quad \quad \quad \quad 1x_{13} \quad \quad \quad \quad \quad + 1x_{23} \geq D_3, \\
 & (a_1 - r_1)x_{11} \quad \quad \quad + (a_2 - r_1)x_{21} \quad \quad \quad \geq 0, \\
 & \quad (b_1 - r_2)x_{12} \quad \quad \quad + (b_2 - r_2)x_{22} \quad \quad \leq 0, \\
 & \quad \quad \quad \quad \quad 1x_{13} \quad \quad \quad \quad \quad - r_3x_{23} \geq 0, \\
 & \quad \quad \quad \quad \quad \quad \quad x_{ij} \geq 0, \quad i = 1, 2, \quad j = 1, 2, 3.
 \end{aligned}$$

3.1.3. Динамическая линейная модель

В каждой из приведенных выше моделей управляемые переменные имеют смысл или объема продукции или объема сырья, произведенных или потребленных за плановый период. Иногда такие статические модели плохо описывают будущее поведение исследуемой системы в будущем, так как каждый из отрезков планового периода может характеризоваться своими, отличными от других отрезков условиями. Например, могут меняться требования к объему выпускаемой продукции, может меняться прибыль, получаемая с единицы продукции, а также могут изменяться другие параметры модели. В этом случае необходимо заменить каждую из управляемых переменных статической модели несколькими управляемыми переменными – по одной на каждый временной отрезок планового периода.

При этом динамическая задача обычно не сводится к ряду задач оптимизации для последовательных периодов времени, рассматриваемых изолированно друг от друга. Например, решая задачу рационального выбора состава комбикорма, управляющий допускает некоторое ослабление требований к составу пищевой смеси в течение одной единицы времени, рассчитывая на соответствующую компенсацию в последующие периоды. Общим для подобных моделей динамического планирования является то, что текущие управляющие решения проявляются как в период, относящийся непосредственно к моменту принятия решения, так и в последующие периоды.

Для описания взаимодействий между отдельными отрезками планового периода вводятся дополнительные управляемые переменные. Примером таких переменных являются объемы запасов, возникшие на t -ых отрезках планового периода ($t = \overline{1, T}$), которые могут быть использованы на последующих отрезках.

В качестве примера рассмотрим последнюю задачу составления жидких смесей с учетом следующих изменений:

а) производятся только 2 продукта: 1 и 2;

б) длительность планового периода равна T , причем объемы химических компонентов, спрос на готовую продукцию, а также доход с единицы продукции зависят от того, какой единичный отрезок планового периода рассматривается. Обозначим соответствующие величины как S_{it}, D_{jt}, P_{ijt} ($i = 1, 2; j = 1, 2; t = \overline{1, T}$);

в) излишки химических компонентов 1 и 2 можно складировать для того чтобы использовать их в последующие временные отрезки. Хранение единицы i -го компонента в течении t -го единичного периода обходится при этом в c_{it} рублей затрат;

г) излишки готовых продуктов 1, 2, произведенных на t -м единичном отрезке, можно складировать, чтобы реализовать их в последующие периоды. При этом хранение единицы j -го продукта в течении t -го единичного периода обходится в r_{jt} рублей затрат.

Построение математической модели начнем с введения управляемых переменных. Во-первых, это x_{ijt} ($i = 1, 2; j = 1, 2; t = \overline{1, T}$). x_{ijt} показывает, сколько литров i -го химического компонента расходуется в t -м периоде для получения j -го продукта. Вторая группа управляемых переменных – y_{it} ($i = 1, 2; t = \overline{1, T-1}$). y_{it} показывает, сколько литров i -го химического компонента находится на складе в конце t -го единичного периода. Последняя группа переменных – z_{jt} ($j = 1, 2, t = \overline{1, T-1}$). z_{jt} показывает, сколько литров j -го готового продукта складировается в конце t -го периода.

Запишем целевую функцию, имеющую смысл прибыли, а также ограничения модели:

$$\begin{aligned}
& \sum_{t=1}^T (p_{11t}x_{11t} + p_{12t}x_{12t} + p_{21t}x_{21t} + p_{22t}x_{22t}) - \sum_{t=1}^{T-1} (c_{1t}y_{1t} + c_{2t}y_{2t} + h_{1t}z_{1t} + h_{2t}z_{2t}) \Rightarrow \max \\
& \begin{array}{rcl}
x_{111} & + x_{121} & + y_{11} \leq S_{11}, \\
& & x_{211} + x_{221} + y_{21} \leq S_{21}, \\
x_{111} & + x_{211} & - z_{11} \geq D_{11}, \\
& x_{121} & + x_{221} - z_{21} \geq D_{21}, \\
(a_1 - r_1)x_{111} & + (a_2 - r_1)x_{211} & \geq 0, \\
& (b_1 - r_2)x_{121} & + (b_2 - r_2)x_{221} \leq 0, \\
& \dots & \dots
\end{array} \\
& \begin{array}{rcl}
x_{11t} & + x_{12t} & + y_{1t} - y_{1,t-1} \leq S_{1t}, \\
& & x_{21t} + x_{22t} + y_{2t} - y_{2,t-1} \leq S_{2t}, \\
x_{11t} & + x_{21t} & - z_{1t} + z_{1,t-1} \geq D_{1t}, \\
& x_{12t} & + x_{22t} - z_{2t} + z_{2,t-1} \geq D_{2t}, \\
(a_1 - r_1)x_{11t} & + (a_2 - r_1)x_{21t} & \geq 0, \\
& (b_1 - r_2)x_{12t} & + (b_2 - r_2)x_{22t} \leq 0, \\
& \dots & \dots
\end{array} \\
& \begin{array}{rcl}
x_{11T} & + x_{12T} & - y_{1,T-1} \leq S_{1T}, \\
& & x_{21T} + x_{22T} - y_{2,T-1} \leq S_{2T}, \\
x_{11T} & + x_{21T} & + z_{1,T-1} \geq D_{1T}, \\
& x_{12T} & + x_{22T} + z_{2,T-1} \geq D_{2T}, \\
(a_1 - r_1)x_{11T} & + (a_2 - r_1)x_{21T} & \geq 0, \\
& (b_1 - r_2)x_{12T} & + (b_2 - r_2)x_{22T} \leq 0,
\end{array} \\
& x_{ijt} \geq 0, \quad i=1,2, \quad j=1,2, \quad t=\overline{1,T}, \\
& y_{it}, z_{jt} \geq 0, \quad i=1,2, \quad j=1,2, \quad t=\overline{1,T-1}.
\end{aligned}$$

Рассмотрим ограничения, относящиеся к t -му временному отрезку. Первые два из них описывают ограниченность поступления химических компонентов из среды на t -ом отрезке. Левая часть каждого из этих неравенств содержит затраты одного из компонентов. Эти затраты обусловлены как нуждами производства на данном этапе ($x_{i1t} + x_{i2t}$), так и увеличением запасов сырья на данном этапе ($y_{it} - y_{i,t-1}$).

Следующие два ограничения описывают необходимость удовлетворения спроса на t -м этапе на готовую продукцию. Левая часть каждого из этих неравенств описывает тот объем соответствующего готового продукта, который на t -м этапе потребляется. Он образуется за счет производства на данном этапе $(x_{1jt}+x_{2jt})$ и за счет уменьшения складских запасов продукции $(z_{j,t-1}-z_{j,t})$.

Отличия в записи ограничений для 1-го и T -го этапов обусловлены тем, что для первого из них нет запасов на начало, а для второго – на конец соответствующего отрезка.

Нетрудно заметить, что если запретить создание запасов сырья и продукции, то данная модель разобьется на T независимых статических моделей.

Подобно задаче составления жидких смесей другие рассмотренные нами статические модели могут быть также положены в основу построения соответствующих динамических моделей. Приведем лишь управляемые переменные этих моделей.

1. Для динамической модели задачи распределения ресурсов:

x_{jt} ($j = \overline{1,4}, t = \overline{1,T}$) - объем продукции, выпускаемой j -м технологическим процессом на t -м интервале времени;

y_{jt} ($j = \overline{1,4}, t = \overline{1,T-1}$) - объем продукции, выпущенной j -м технологическим процессом и хранящийся на складе в конце t -го отрезка времени;

Z_{it} ($i = \overline{1,3}, t = \overline{1,T-1}$) - объем i -го вида сырья, хранящегося на складе в конце t -го отрезка времени.

2. Для динамической модели задачи загрузки оборудования:

x_{ijt} ($i = 1,2, j = 1,2,3, t = \overline{1,T}$) - количество станков типа i , занятых изготовлением ткани T_j на отрезке времени t ;

y_{jt} ($j = 1,2,3, t = \overline{1,T-1}$) - количество ткани T_j , хранящейся на складе в конце отрезка t .

3. Для динамической модели составления комбикорма:

x_{jt} ($j = 1,2,3, t = \overline{1,T}$) - количество зерна типа j , закупаемого на отрезке t ;

y_{jt} ($j = 1,2,3, t = \overline{1,T-1}$) - запас зерна типа j на конец отрезка t ;

Z_{it} ($i = \overline{1,4}, t = \overline{1,T-1}$) - запас (излишек) питательного вещества i на конец отрезка t .

3.2 Предварительное преобразование линейной модели

Выявленная на этапе построения математической модели линейная оптимизационная модель включает целевую функцию вида:

$$c_1x_1 + c_2x_2 + \dots + c_nx_n,$$

значение которой в зависимости от содержания задачи должно быть максимизировано или минимизировано при наличии ограничений на управляемые переменные $x_j (j = \overline{1, n})$, каждое из которых относится к одному из следующих типов:

$$\begin{aligned} a_1x_1 + a_2x_2 + \dots + a_nx_n &\geq a, \\ b_1x_1 + b_2x_2 + \dots + b_nx_n &= b, \\ d_1x_1 + d_2x_2 + \dots + d_nx_n &\leq d. \end{aligned}$$

Кроме того, могут иметься ограничения $x_j \geq 0$.

Для того чтобы применить для нахождения решения данной модели стандартный симплексный алгоритм, необходимо представить модель в стандартной (канонической) форме. Вообще говоря, существуют несколько канонических форм записи моделей линейного программирования. Приведем две наиболее распространенные:

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

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

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

$$2) \sum_{j=1}^n c_j x_j \Rightarrow \min,$$

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

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

Рассмотрим приемы, пользуясь которыми любую линейную оптимизационную модель можно привести к любой из этих канонических форм.

1. Переход от максимизации к минимизации и наоборот. Данный переход осуществим всегда. Для этого одновременно с изменением знака оптимизации необходимо произвести изменения знаков перед всеми коэффициентами в целевой функции. Например, максимизация $\sum_{j=1}^n c_j x_j$ эквивалентна ми-

минимизации $\sum_{j=1}^n (-c_j)x_j$. Если при этом V – оптимальное значение линейной функции $\sum_{j=1}^n (-c_j)x_j$, то $-V$ представляет собой оптимальное значение для $\sum_{j=1}^n c_j x_j$.

2. Инвертирование неравенств. Каждое из неравенств можно инвертировать, если учесть, что $\sum_{j=1}^n a_{ij}x_j \leq b_i$ эквивалентно $\sum_{j=1}^n (-a_{ij})x_j \geq -b_i$. Например, неравенство $x_1 - x_2 \leq -6$ эквивалентно неравенству $-x_1 + x_2 \geq 6$.

3. Обращение неравенств в равенства. Любое неравенство можно представить в виде равенства, введя новую неотрицательную переменную следующим образом:

$$\sum_{j=1}^n a_{ij}x_j \leq b_i \text{ эквивалентно } \sum_{j=1}^n a_{ij}x_j + s = b_i, \text{ где } s \geq 0,$$

$$\sum_{j=1}^n a_{ij}x_j \geq b_i \text{ эквивалентно } \sum_{j=1}^n a_{ij}x_j - t = b_i, \text{ где } t \geq 0.$$

Переменная s называется *остаточной переменной*, переменная t называется *избыточной переменной*. Например, в п.3.1.1 ограничение на затраты материала Y имело вид:

$$10x_1 + 8x_2 + 6x_3 + 5x_4 \leq 100,$$

что эквивалентно равенству

$$10x_1 + 8x_2 + 6x_3 + 5x_4 + y = 100, \text{ где } y \geq 0.$$

Здесь y есть остаток, т.е. неиспользованная часть материала Y .

Аналогично, для модели в п.3.1.2, ограничение, относящееся к веществу A , можно записать в виде:

$$4x_1 + 6x_2 + 14x_3 - z = 2500, \text{ где } z \geq 0.$$

Здесь z есть мера превышения потребности в веществе A , т.е. z является избыточной переменной.

3. Обращение равенств в неравенства. Любое линейное уравнение, а также любую систему линейных уравнений можно представить в виде совокупности линейных неравенств с помощью одного дополнительного ограничения. Например, равенство $x=10$ эквивалентно комбинации неравенств $x \leq 10$ и $x \geq 10$, которая в свою очередь может быть записана в виде пары неравенств: $x \leq 10$ и $-x \leq -10$.

Система уравнений $x=2, y=4$ эквивалентна комбинации неравенств $x \leq 2, y \leq 4, x+y \geq 6$, которую в свою очередь можно представить в виде: $x \leq 2, y \leq 4, -x-y \leq -6$. Изложенные соображения допускают обобщение: систему уравнений $\sum_{j=1}^n a_{ij}x_j = b_i$ ($i = \overline{1, m}$) можно записать в виде:

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

где $\alpha_j = -\sum_{i=1}^m a_{ij}$, $\beta = -\sum_{i=1}^m b_i$.

В качестве примера рассмотрим систему уравнений:

$$\begin{aligned} x_1 + x_2 &= 1 \\ 2x_1 - 4x_3 &= -5. \end{aligned}$$

Эта система эквивалентна следующей системе неравенств:

$$\begin{aligned} x_1 + x_2 &\leq 1, \\ 2x_1 - 4x_3 &\leq -5, \\ -3x_1 - x_2 + 4x_3 &\leq 4. \end{aligned}$$

4. Переход от переменных, не имеющих ограничений в знаке, к неотрицательным переменным. Допустим, что переменная x_j не ограничена в знаке. В этом случае x_j полагается разности двух неотрицательных переменных: $x_j = x'_j - x''_j$, где $x'_j \geq 0$, $x''_j \geq 0$. Далее данная разность подставляется всюду в модели вместо x_j .

5. Переход от переменных, значения которых ограничены снизу, к неотрицательным переменным. Когда x_j ограничена снизу некоторой константой $b_j \neq 0$, x_j можно представить в виде: $x_j = b_j + x'_j$, где $x'_j \geq 0$. Далее данная сумма подставляется всюду в модели вместо x_j .

3.3. Графическая интерпретация линейных моделей

Графическая интерпретация элементов линейной модели очень полезна для более глубокого понимания свойств таких моделей. В качестве примера рассмотрим модель:

$$\begin{aligned} 7x_1 + 14x_2 &\Rightarrow \max, \\ 5x_1 + 10x_2 &\leq 20, \\ 6x_1 + 3x_2 &\leq 12, \\ x_1 &\geq 0, x_2 \geq 0. \end{aligned}$$

Эта же модель графически представлена на рис. 11. На этом рисунке многоугольник Oabc есть область допустимых значений x_1 и x_2 . Иными словами это графическое изображение множества решений. Данное множество является **выпуклым**, так как любой отрезок, соединяющий любые две точки многоугольника Oabc, лежит или внутри его, или на границе. Вершины O, a, b, c называются **экстремальными точками**. Они не могут принадлежать внутренней части ни одного из отрезков, соединяющих две различные точки рассматриваемого множества.

Параллельные прямые на рис.11 соответствуют линиям равного уровня целевой функции. При этом стрелка показывает направление увеличения значения целевой функции. Двигая линию равного уровня в этом направлении, нетрудно убедиться, что наибольшее (оптимальное) решение соответствует экстремальной точке b . В этой точке $x_1 = \frac{4}{3}$, $x_2 = \frac{4}{3}$, $7x_1 + 14x_2 = 28$.

Если в выражении для целевой функции изменить коэффициенты, угол наклона параллельных прямых по отношению к оси абсцисс будет другим и точка, соответствующая оптимальному решению, может переместиться. Однако всегда существует оптимальное решение, соответствующее некоторой экстремальной точке.

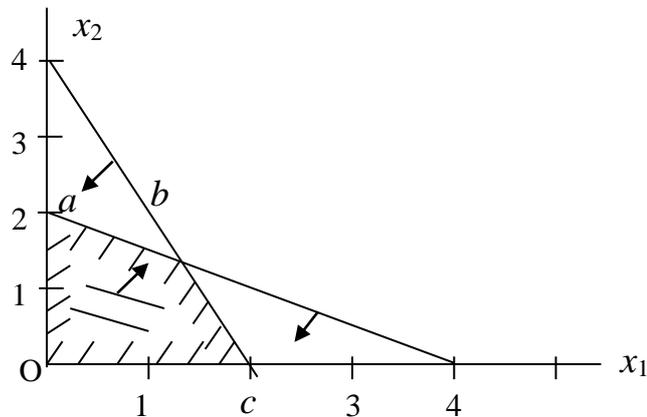


Рис. 11. Графическое представление модели с конечным решением

Пусть ограничения модели останутся те же самые, а целевая функция будет другой: $10x_1 + 20x_2 \Rightarrow \max$. В этом случае все точки на отрезке ab , а их бесконечное множество, являются оптимальными. Следовательно, решение $x_1 = \frac{4}{3}$, $x_2 = \frac{4}{3}$ по-прежнему оптимально. Но теперь оптимальным является и решение $x_1=0$, $x_2=2$. То же самое можно сказать и о всех других точках отрезка ab .

Рассмотрим следующую модель:

$$\begin{aligned} -6x_1 + 18x_2 &\Rightarrow \max, \\ -x_1 - x_2 &\leq -4, \\ -2x_1 + 2x_2 &\leq 2, \\ x_1 &\geq 0, x_2 \geq 0. \end{aligned}$$

Графически модель представлена на рис.12, из которого видно, что задача имеет неограниченно большое значение целевой функции, которое достигается при бесконечно больших допустимых значениях x_1 и x_2 .

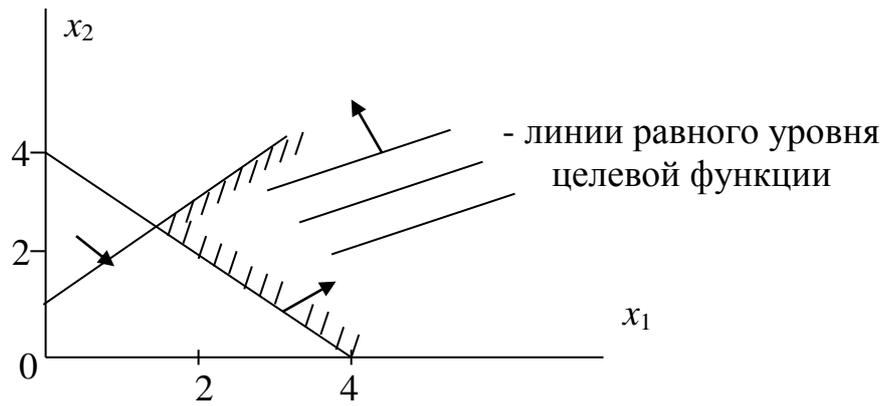


Рис.12. Графическое представление модели с бесконечным решением

Рассмотрим пример модели, не имеющей решения:

$$\begin{aligned} x_1 + x_2 &\Rightarrow \max, \\ -x_1 + x_2 &\leq -1, \\ x_1 - x_2 &\leq -1, \\ x_1 &\geq 0, x_2 &\geq 0. \end{aligned}$$

Графическое представление данной модели приведено на рис.13, из которого видно, что множество решений пустое.

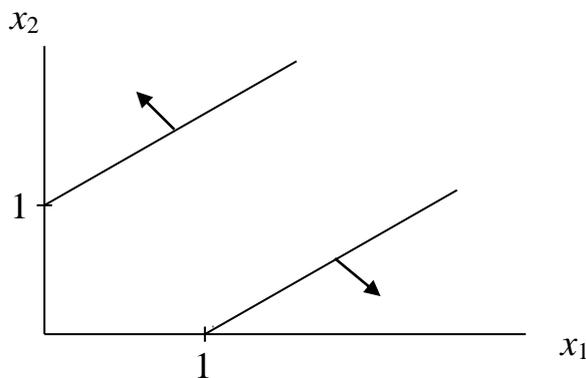


Рис.13. Графическое представление модели, не имеющей допустимых решений

Из приведенных графических интерпретаций можно сделать следующие выводы:

- 1) если множество решений является непустым, то оно выпукло и может быть или ограниченным, или неограниченным;
- 2) если множество решений является непустым, то оптимальное значение целевой функции может быть или конечным, или неограниченно большим.

Рассмотрим теперь графическое (а точнее - геометрическое) представление моделей с n ($n > 2$) управляемыми переменными. Пусть модель имеет вид:

$$\sum_{j=1}^n c_j x_j \Rightarrow \max, \quad (3.6)$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = \overline{1, m}), \quad (3.7)$$

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

Представим себе n -мерное евклидово пространство, т.е. множество точек, каждая из которых задается с помощью n координат (x_1, x_2, \dots, x_n) . Так как $x_j \geq 0$, рассмотрению подлежит только положительная область n -мерного евклидова пространства.

Каждому уравнению

$$\sum_{j=1}^n a_{ij} x_j = b_i \quad (i = \overline{1, m}) \quad (3.8)$$

соответствует гиперплоскость, которая делит рассматриваемое пространство на два полупространства, в одном из которых соответствующее i -е неравенство (3.7) выполняется, а в другом не выполняется. Совокупность m гиперплоскостей, соответствующих набору уравнений (3.8), а также n гиперплоскостей $x_j = 0$ вырезают из n -мерного пространства решений область допустимых решений. В случае, если это множество непустое, оно является выпуклым и **полиэдральным**. При этом грани полиэдра лежат на гиперплоскостях (3.8) и $x_j = 0$, а ребра и вершины лежат на их пересечениях.

Если допустим, что целевая функция равна какому-то числу, то приравняв (3.6) этому числу, получим уравнение гиперплоскости в n -мерном пространстве решений, в каждой точке которой целевая функция равна данному числу. Гиперплоскости, соответствующие различным значениям целевой функции, параллельны друг другу. Отсюда следует, что никакой точке, лежащей строго внутри полиэдра допустимых решений, не может соответствовать оптимальное решение, так как всегда найдутся точки, которые лежат на гиперплоскостях, соответствующих большим значениям целевой функции.

Если оптимальное значение целевой функции конечное, то оптимальное решение обязательно находится в экстремальной точке полиэдра допустимых решений. Если же n экстремальных точек являются оптимальными, то оптимальными являются также все точки, лежащие на ребрах и гранях, соединяющих эти экстремальные точки.

Из геометрической интерпретации следует очень простой алгоритм, который может показаться пригодным для поиска оптимального решения. Так как известно, что оптимальное решение находится в экстремальной точке полиэдра допустимых решений, то для его определения достаточно найти все экстремальные точки полиэдра, а затем выбрать из них наилучшую с точки зрения достижения экстремума целевой функции. При поиске экстремальных

точек следует учесть, что в каждой из них пересекаются n из $(n+m)$ граничных гиперплоскостей.

Каждая из точек пересечения не обязательно является экстремальной точкой полиэдра. Чтобы проверить, является ли точка пересечения n гиперплоскостей экстремальной, достаточно проверить, удовлетворяет ли она неравенствам, соответствующим остальным m гиперплоскостям. Существенным недостатком данного алгоритма, который делает невозможным его практическое применение, является слишком большое число точек пересечения гиперплоскостей, равное C_{n+m}^n . Например, если $n=15$, $m=10$, число точек пересечения равно $C_{15+10}^{15} = \frac{(n+m)!}{m!n!} = \frac{25!}{15!10!} = 3.268.760$.

Поэтому для поиска оптимального решения нужно использовать такой алгоритм, при выполнении которого рассматривается только часть экстремальных точек. Таким алгоритмом является симплексный алгоритм.

3.4 Симплексный алгоритм

Геометрическая интерпретация симплексного алгоритма заключается в следующем. Берется любая экстремальная точка полиэдра допустимых решений в качестве исходной точки. Далее рассматриваются соседние экстремальные точки, которые соединены с исходной точкой ребрами полиэдра. Из этих точек выбирается та точка, которой соответствует лучшее значение целевой функции. Далее рассматриваются точки, соседние новой точке, и так далее до тех пор, пока не прекратится процесс улучшения целевой функции. Полученной точке соответствует оптимальное значение целевой функции.

Симплексный алгоритм состоит из следующих шагов:

Шаг 1. Получение исходного допустимого решения. Для этого выбираются m ненулевых переменных, затем эти переменные исключаются из выражения для целевой функции.

Шаг 2. Делается проверка, нельзя ли за счет одной из переменных, приравненных вначале к нулю, улучшить значение целевой функции, придавая переменной положительные значения. Если это возможно, переход к шагу 3. В противном случае вычисления закончены.

Шаг 3. Находится предельное значение выбранной на шаге 2 переменной. Увеличение значения этой переменной допустимо до тех пор, пока одна из m переменных, вошедших ранее в пробное решение не обратится в нуль. Выбранная переменная исключается из выражения для целевой функции и вводится в пробное решение.

Шаг 4. Система m уравнений разрешается относительно переменных, вошедших в новое пробное решение. Эти переменные исключаются из выражения для целевой функции. Возврат к шагу 2.

Более подробно симплексный алгоритм рассмотрим на примере.

Пример. Пусть дана линейная оптимизационная модель:

$$\begin{aligned} 3x_1 + 4x_2 + 7x_3 + 9x_4 &\Rightarrow \max, \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 &\leq 20, \\ 5x_1 + 3x_2 + x_3 + x_4 &\leq 100, \\ 5x_1 + 5x_2 + 10x_3 + 20x_4 &\leq 80, \\ x_j &\geq 0 \quad (j=1,2,3,4). \end{aligned}$$

Шаг 1.

Обозначим через x_0 значение целевой функции и введем в рассмотрение остаточные переменные x_5, x_6, x_7 . В результате получим систему уравнений:

$$\begin{aligned} x_0 - 3x_1 - 4x_2 - 7x_3 - 9x_4 &= 0 \quad (\text{строка } 0), \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 + x_5 &= 20 \quad (\text{строка } 1), \\ 5x_1 + 3x_2 + x_3 + x_4 + x_6 &= 100 \quad (\text{строка } 2), \\ 5x_1 + 5x_2 + 10x_3 + 20x_4 + x_7 &= 80 \quad (\text{строка } 3), \end{aligned}$$

где все переменные могут принимать лишь неотрицательные значения. Введение переменной x_0 позволяет записать выражение для целевой функции в виде уравнения.

Требуется выбрать первоначальное допустимое решение записанной системы уравнений. Удобнее всего начать с $x_0=0, x_5=20, x_6=100, x_7=80$ при нулевых значениях остальных переменных. То есть первое решение строится только из свободных (остаточных) переменных. Назовем его исходным базисным (опорным) решением, а переменные x_0, x_5, x_6, x_7 – базисными переменными или сокращенно базисом. Остальные переменные называются небазисными.

Термин **базис** отражает то обстоятельство, что базисное решение содержит столько переменных, сколько линейных соотношений содержит рассматриваемая модель (в примере $m=4$). Причем значения этих четырех переменных однозначно определяются положительными константами в правых частях уравнений. Из геометрической интерпретации симплексного алгоритма следует, что термины **базисное решение** и **экстремальные точки полиэдра** модели означают одно и то же.

Покажем, что любой базис (координаты экстремальной точки) содержит не более m неотрицательных переменных из общего числа переменных $n+m$. Это следует из того, что в каждой экстремальной точке пересекаются m граничных гиперплоскостей из $m+n$. Каждой из m пересекаемых плоскостей соответствует нулевое значение соответствующей свободной переменной (если

плоскость относится к первым m ограничениям) или нулевое значение управляемой переменной (для плоскости $x_j=0$).

Итерация 1

Шаг 2. Так как нам нужно обеспечить наибольшее возможное значение x_0 , то выбранное пробное решение является явно плохим. Но его, несомненно, можно улучшить. Обратим внимание на строку 0. Во-первых, в нее входят, за исключением x_0 , только небазисные переменные. Во-вторых, каждый коэффициент в строке 0 определяет положительное (если перед ним стоит минус) или отрицательное (если перед ним стоит плюс) приращение x_0 при увеличении на единицу соответствующей базисной переменной.

На шаге 2 ищется переменная, за счет которой можно улучшить значение целевой функции путем включения ее в базис. Выбор переменной осуществляется согласно следующему правилу: если в строке 0 имеются небазисные переменные, коэффициенты при которых отрицательны, следует выбрать переменную (обозначим ее через x_j) с наибольшим абсолютным значением стоящего перед ней коэффициента, то есть ту переменную, которая обеспечивает наибольшее удельное приращение значения целевой функции. Если все небазисные переменные строки 0 имеют положительные или нулевые коэффициенты, оптимальное решение уже получено.

В соответствии с данным правилом в базис следует ввести x_4 . Каждое единичное приращение x_4 приводит к увеличению x_0 на 9.

Шаг 3. Ищется предельное значение переменной, за счет которой можно улучшить значение целевой функции. Понятно, что чем больше x_4 , тем больше x_0 . Но беспредельно расти x_4 не может, так как этому препятствуют уравнения в строках 1,2,3. В каждой из этих строк x_4 растет за счет соответствующей базисной переменной. Наименьший рост x_4 позволяет строка 3: до $\frac{80}{20} = 4$, при этом x_7 обращается в 0.

В общем случае справедливо правило: рассмотрим отношения чисел, стоящих в правых частях уравнений, кроме нулевой строки, к соответствующим коэффициентам при новой базисной переменной x_j (не обращая внимание на отношения, в которых знаменатель равен нулю или представляет отрицательное число). Наименьшее из этих отношений и есть значение новой переменной x_j в очередном пробном решении. Та прежняя базисная переменная x_k , которой соответствует наименьшее отношение, исключается из базиса, то есть следует положить $x_k=0$. Если ни в одном из отношений знаменатель не является положительным числом, то имеем неограниченное оптимальное решение, так как x_j может быть взято сколько угодно большим.

Теперь, когда известно, что x_7 следует заменить в базисе на x_4 , перейдем к шагу 4.

Шаг 4. Перепишем систему уравнений таким образом, чтобы в строке 3 коэффициент при x_4 был равен единице, а в строках 0,1,2 – нулю. Процедуру,

с помощью которой это достигается, называют *операцией замены базиса* или *операцией замены опорного плана*.

Сначала разделим обе части уравнения в строке 3 на коэффициент при x_4 , то есть на 20:

$$\begin{aligned} x_0 - 3x_1 - 4x_2 - 7x_3 - 9x_4 &= 0 \quad (\text{строка } 0), \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 + x_5 &= 20 \quad (\text{строка } 1), \\ 5x_1 + 3x_2 + x_3 + x_4 + x_6 &= 100 \quad (\text{строка } 2), \\ \frac{1}{4}x_1 + \frac{1}{4}x_2 + \frac{1}{2}x_3 + x_4 + \frac{1}{20}x_7 &= 4 \quad (\text{строка } 3), \end{aligned}$$

В результате коэффициент при x_4 в строке 3 равен 1. Обратим в нули коэффициенты при x_4 в строках 0, 1 и 2, действуя следующим образом:

- умножим строку 3 на 9 и результат прибавим к строке 0;
- умножим строку 3 на (-2) и результат прибавим к строке 1;
- умножим строку 3 на (-1) и результат прибавим к строке 2.

В результате получим:

$$\begin{aligned} x_0 - \frac{3}{4}x_1 - \frac{7}{4}x_2 - \frac{5}{2}x_3 + \frac{9}{20}x_7 &= 36 \quad (\text{строка } 0), \\ \frac{3}{2}x_1 + \frac{3}{2}x_2 + x_3 + x_5 - \frac{1}{10}x_7 &= 12 \quad (\text{строка } 1), \\ \frac{19}{4}x_1 + \frac{11}{4}x_2 + \frac{1}{2}x_3 + x_6 - \frac{1}{20}x_7 &= 96 \quad (\text{строка } 2), \\ \frac{1}{4}x_1 + \frac{1}{4}x_2 + \frac{1}{2}x_3 + x_4 + \frac{1}{20}x_7 &= 4 \quad (\text{строка } 3), \end{aligned}$$

Удобство полученного представления линейной модели в том, что из нее сразу видны значения переменных нового базиса: $x_0 = 36$, $x_5 = 12$, $x_6 = 96$, $x_4 = 4$. Прибыль x_0 увеличилась на первой итерации с 0 до 36 за счет новой базисной переменной x_4 : $36 = 4 \cdot 9$.

Итерация 2

Шаг 2. В базис следует ввести x_3 .

Шаг 3. Ищем отношения правых частей уравнений к соответствующим коэффициентам при x_3 . Они равны: $12/1$, $192/11$, $8/1$. Следовательно, из базиса должен быть исключен x_4 .

Шаг 4. Вначале выполним нормировку коэффициента при x_3 в строке 3:

$$\begin{aligned}
x_0 - \frac{3}{4}x_1 - \frac{7}{4}x_2 - \frac{5}{2}x_3 &+ \frac{9}{20}x_7 = 36 \quad (\text{строка } 0), \\
\frac{3}{2}x_1 + \frac{3}{2}x_2 + x_3 &+ x_5 - \frac{1}{10}x_7 = 12 \quad (\text{строка } 1), \\
\frac{19}{4}x_1 + \frac{11}{4}x_2 + \frac{1}{2}x_3 &+ x_6 - \frac{1}{20}x_7 = 96 \quad (\text{строка } 2), \\
\frac{1}{2}x_1 + \frac{1}{2}x_2 + x_3 + 2x_4 &+ \frac{1}{10}x_7 = 8 \quad (\text{строка } 3),
\end{aligned}$$

Исключим x_3 из уравнений в строках 0, 1 и 2, выполнив действия:

- а) умножим строку 3 на $5/2$ и результат сложим со строкой 0;
- б) умножим строку 3 на (-1) и результат сложим со строкой 1;
- в) умножим строку 3 на $(-1/2)$ и результат сложим со строкой 2.

В результате получим:

$$\begin{aligned}
x_0 + \frac{1}{2}x_1 - \frac{1}{2}x_2 &+ 5x_4 + \frac{7}{10}x_7 = 56 \quad (\text{строка } 0), \\
x_1 + x_2 &- 2x_4 + x_5 - \frac{1}{5}x_7 = 4 \quad (\text{строка } 1), \\
\frac{9}{2}x_1 + \frac{5}{2}x_2 &- x_4 + x_6 - \frac{1}{10}x_7 = 92 \quad (\text{строка } 2), \\
\frac{1}{2}x_1 + \frac{1}{2}x_2 + x_3 + 2x_4 &+ \frac{1}{10}x_7 = 8 \quad (\text{строка } 3),
\end{aligned}$$

Третий пробный базис имеет вид: $x_0 = 56, x_5 = 4, x_6 = 92, x_3 = 8$.

Итерация 3

Шаг 2. В базис следует ввести x_2 .

Шаг 3. Ищем отношения правых частей уравнений к соответствующим коэффициентам при x_2 . Они равны: 4 (строка 1), $92 \cdot 2/5 = 36,8$ (строка 2) и 16 (строка 3). Следовательно, из базиса должен быть исключен x_5 .

Шаг 4. Нормировать коэффициенты при x_2 в строке 1 нет необходимости. Теперь исключим x_2 из уравнений в строке 0, 2 и 3, выполнив действия:

- а) умножим строку 1 на $1/2$ и результат сложить со строкой 0;
- б) умножим строку 1 на $(-5/2)$ и результат сложить со строкой 2;
- в) умножим строку 1 на $(-1/2)$ и результат сложить со строкой 3.

В результате получим:

$$\begin{array}{rcll}
 x_0 + x_1 & +4x_4 + \frac{1}{2}x_5 & +\frac{3}{5}x_7 = 58 & (\text{строка } 0), \\
 x_1 + x_2 & -2x_4 + x_5 & -\frac{1}{5}x_7 = 4 & (\text{строка } 1), \\
 2x_1 & +4x_4 - \frac{5}{2}x_5 + x_6 & +\frac{2}{5}x_7 = 82 & (\text{строка } 2), \\
 x_3 + x_4 & -\frac{1}{2}x_5 & +\frac{1}{5}x_7 = 6 & (\text{строка } 3),
 \end{array}$$

Итерация 4

Шаг 2. Так как в строке 0 все коэффициенты положительны, получено оптимальное решение: $x_0 = 58$, $x_2 = 4$, $x_6 = 82$, $x_3 = 6$.

3.5. Получение исходного базиса

Так как в модели, решенной в п. 3.4 симплексным методом, каждое из ограничений имеет вид: $\sum_{j=1}^n a_{ij}x_j \leq b_i$, где $b_i \geq 0$, выполнение шага 1 симплексного алгоритма (то есть построение первого пробного решения) производится очень просто путем ввода остаточных переменных. Если в модели есть линейные ограничения, которые после приведения к виду неравенств (\leq) имеют в правой части отрицательные константы, то получение исходного базиса не так просто и включает следующие шаги.

Шаг 1. Запись всех ограничений в виде равенств типа:

$$\sum_{j=1}^n a_{ij}x_j = b_i \quad (i = \overline{1, m}), \quad b_i \geq 0. \quad (3.9)$$

Подобная запись осуществима всегда.

Шаг 2. Для каждого из равенств (3.9), которое имеет в левой части некоторую переменную с коэффициентом +1, входящую только в данное равенство, как это имеет место, например, в случае остаточных переменных, соответствующая переменная включается в первый пробный базис. Для каждого из равенств в (3.9), у которой нет такой переменной, вводится искусственная переменная $y_i \geq 0$:

$$\sum_{j=1}^n a_{ij}x_j + y_i = b_i, \quad b_i \geq 0.$$

Все переменные y_i также включаются в исходный базис.

Шаг 3. К подлежащей максимизации целевой функции прибавляется сумма всех искусственных переменных y_i с общим коэффициентом, значение которого выбирается достаточно большим:

$$x_0 - \sum_{j=1}^n c_j x_j + \sum_{i=1}^m M y_i = 0.$$

Коэффициент M имеет смысл отрицательной удельной прибыли. Его назначение в том, чтобы обязательно обеспечить в процессе выполнения симплексного алгоритма изгнание из базиса всех искусственных переменных.

Далее симплексный алгоритм выполняется обычным образом.

Пример. Дана модель:

$$-6x_1 - 4x_2 \Rightarrow \max,$$

$$x_1 + x_2 = 20,$$

$$x_1 \geq 10,$$

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

Шаг 1. После добавления избыточной переменной x_3 с коэффициентом -1 модель примет вид:

$$x_0 + 6x_1 + 4x_2 = 0 \text{ (строка 0)},$$

$$x_1 + x_2 = 20 \text{ (строка 1)},$$

$$x_1 - x_3 = 10 \text{ (строка 2)}$$

Шаг 2. Для равенства в строке 1 в качестве базисной переменной удобно взять x_2 . Для равенства в строке 2 необходимо ввести искусственную переменную y_1 :

$$x_0 + 6x_1 + 4x_2 = 0 \text{ (строка 0)},$$

$$x_1 + x_2 = 20 \text{ (строка 1)},$$

$$x_1 - x_3 + y_1 = 10 \text{ (строка 2)}.$$

Шаг 3. Добавим в целевую функцию переменную y_1 с большим отрицательным коэффициентом, например, равным -10 :

$$x_0 + 6x_1 + 4x_2 + 10y_1 = 0 \text{ (строка 0)},$$

$$x_1 + x_2 = 20 \text{ (строка 1)},$$

$$x_1 - x_3 + y_1 = 10 \text{ (строка 2)}.$$

Перепишем все равенства так, чтобы в каждом из них была только своя базисная переменная. Для этого сначала вычтем из строки 0 строку 1, умноженную на 4:

$$\begin{aligned} x_0 + 2x_1 + 10y_1 &= -80 \text{ (строка 0),} \\ x_1 + x_2 &= 20 \text{ (строка 1),} \\ x_1 - x_3 + y_1 &= 10 \text{ (строка 2).} \end{aligned}$$

Затем вычтем из строки 0 строку 2, умноженную на 10:

$$\begin{aligned} x_0 - 8x_1 + 10x_3 &= -180 \text{ (строка 0),} \\ x_1 + x_2 &= 20 \text{ (строка 1),} \\ x_1 - x_3 + y_1 &= 10 \text{ (строка 2).} \end{aligned}$$

Таким образом, получили следующий исходный базис: $x_0=-180$, $x_2=20$, $y_1=10$.

В настоящее время существует достаточно большое число готовых программ, выполняющих реализацию симплексного алгоритма. Размерность решаемых задач для них может быть очень большой – до 1000 ограничений. С ростом числа m ограничений модели, во-первых, увеличивается число итераций симплексного алгоритма, которое обычно составляет от $1,5m$ до $3m$. Во-вторых, естественно, увеличивается объем вычислений на каждой итерации. Из опыта известно, что общая трудоемкость вычислений, которую можно измерить в единицах затрат машинного времени, расчет пропорционально кубу числа ограничений. Поэтому задача с 200-ми ограничениями, вероятно, будет решаться на ЭВМ в 8 раз дольше, чем со 100 ограничениями.

3.6. Анализ моделей на чувствительность и двойственная задача

3.6.1. Введение

В большинстве случаев кроме оптимизации решения, ЛПР хочет знать, в каком интервале можно менять параметры в модели без существенного отклонения от найденного оптимума и без значительного нарушения структуры оптимального базиса. Исследование, позволяющее ответить на эти вопросы, называется *анализом линейной модели на чувствительность*, или *анализом линейной модели при известном оптимальном решении*.

На многие вопросы, связанные с анализом на чувствительность, можно ответить, располагая численными данными на заключительной симплексной итерации. Ниже описываются два метода анализа на чувствительность, применимые к задачам вида:

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

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

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

Сущность обоих методов будет полностью показана на рассмотренном ранее примере распределения ресурсов. Запишем для этой задачи исходную и заключительную системы уравнений:

$$\begin{aligned} x_0 - 3x_1 - 4x_2 - 7x_3 - 9x_4 &= 0 \quad (\text{строка } 0), \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 + x_5 &= 20 \quad (\text{строка } 1), \\ 5x_1 + 3x_2 + x_3 + x_4 + x_6 &= 100 \quad (\text{строка } 2), \\ 5x_1 + 5x_2 + 10x_3 + 20x_4 + x_7 &= 80 \quad (\text{строка } 3). \end{aligned} \quad (\text{I})$$

и

$$\begin{aligned} x_0 + x_1 + 4x_4 + \frac{1}{2}x_5 + \frac{3}{5}x_7 &= 58 \quad (\text{строка } 0), \\ x_1 + x_2 - 2x_4 + x_5 - \frac{1}{5}x_7 &= 4 \quad (\text{строка } 1), \\ 2x_1 + 4x_4 - \frac{5}{2}x_5 + x_6 + \frac{2}{5}x_7 &= 82 \quad (\text{строка } 2), \\ x_3 + x_4 - \frac{1}{2}x_5 + \frac{1}{5}x_7 &= 6 \quad (\text{строка } 3). \end{aligned} \quad (\text{F})$$

где x_0 подлежит максимизации.

3.6.2. Изменение целевой функции

Предположим, что у нас изменились некоторые коэффициенты целевой функции. Понятно, что полученное ранее оптимальное решение по-прежнему осталось допустимым. Но оно может быть уже не оптимальным.

Рассмотрим коэффициент при небазисных переменных x_1 и x_4 в строке 0 системы уравнений (F). Интуитивно ясно, что если уменьшить их удельную прибыльность, прежний заключительный базис останется оптимальным. Однако если прибыльность x_1 и x_4 получит достаточно большое положительное приращение, оптимальный базис может смениться.

Предположим, что коэффициент при x_1 получает положительное приращение δ , т.е. становится равным $3+\delta$. Тогда строка 0 системы уравнений (I) примет вид: $x_0 - (3+\delta)x_1 - 4x_2 - 7x_3 - 9x_4 = 0$.

При выполнении каждой итерации симплексного алгоритма мы прибавляли к строке 0 одну из остальных строк, предварительно умножив послед-

ною на некоторую константу. Следовательно, на заключительной итерации строка 0 системы уравнений (F) запишется в виде:

$$x_0 + (1 - \delta)x_1 + 4x_4 + \frac{1}{2}x_5 + \frac{3}{5}x_7 = 58,$$

так как член $-\delta x_1$ сохраняется в строке 0 на любой итерации.

Если $\delta > 1$, то коэффициент при x_1 примет отрицательное значение. В этом случае прежнее решение уже не является оптимальным, так как в базис следует включить x_1 . Аналогично, если бы коэффициент при x_4 увеличился бы на значение, превышающее 4, то прежнее решение перестало бы быть оптимальным. Следовательно, коэффициенты при небазисных переменных в строке 0 на этапе заключительной итерации показывают, в каких пределах могут принимать положительные приращения соответствующие коэффициенты в выражении для целевой функции без нарушения оптимальности ранее полученного решения.

Предположим, что существенно понизилась прибыльность переменной x_2 или x_3 , каждая из которых входит в базис. Вполне возможно, что найденное базисное решение окажется при этом неоптимальным. Оказывается, что и при увеличении коэффициентов при x_2 и x_3 ранее полученное решение может также оказаться неоптимальным.

Ответим на вопрос: в каких пределах можно менять коэффициент при x_2 не нарушая оптимальности найденного ранее решения. Для ответа на этот вопрос выполним операции, аналогичные только что проделанным. При этом строка 0 системы уравнений (I) примет вид: $x_0 - 3x_1 - (4 + \delta)x_2 - 7x_3 - 9x_4 = 0$, а строка 0 в (F) запишется в виде:

$$x_0 + x_1 - \delta x_2 + 4x_4 + \frac{1}{2}x_5 + \frac{3}{5}x_7 = 58.$$

Так как считаем, что x_2 по-прежнему входит в базис, необходимо исключить x_2 из строки 0 в системе (F). Для этого умножим строку 1 в (F) на δ и прибавим к строке 0. В результате строка 0:

$$x_0 + (1 + \delta)x_1 + (4 - 2\delta)x_4 + \left(\frac{1}{2} + \delta\right)x_5 + \left(\frac{3}{5} - \frac{1}{5}\delta\right)x_7 = 58 + 4\delta.$$

Отсюда следует, что при выполнении условия $-1 \leq \delta \leq 2$ полученное решение остается оптимальным. При $\delta < -1$ коэффициент при x_1 принимает отрицательное значение. При $\delta > 2$ отрицательным становится коэффициент при x_4 .

Рассмотренный прием анализа на чувствительность годен и в случае одновременного изменения коэффициентов при нескольких базисных переменных. При этом в строку 0 на последней итерации войдет несколько переменных приращений δ_j , каждое из которых влияет на способ выбора нового базиса. Чтобы базис остался оптимальным, δ_j должны удовлетворять системе неравенств, каждое из которых есть условие неотрицательности коэффициентов при соответствующей небазисной переменной в строке 0.

3.6.3. Изменение констант в правых частях ограничений

Их изменение может привести к тому, что прежний оптимальный базис будет недопустимым. Если он останется допустимым, то он по-прежнему будет оптимальным, так как коэффициенты в строке 0 не меняются.

Рассмотрим правую часть строки 2 системы (I). Произведем замену $100 \rightarrow 100 + \delta$. Заметим, что остаточная переменная в данной строке x_6 входит в базис и на первой и на последней итерации, то есть с коэффициентом $+1$. Поэтому δ сохранится в правой части строки 2 без изменений. Таким образом, ранее полученное решение останется допустимым, если $\delta \geq -82$ (см. строку 2 в F). Иначе x_6 будет < 0 , что недопустимо.

Произведем теперь замену $20 \rightarrow 20 + \delta$ в правой части строки 1 системы уравнений (I). При каких значениях δ полученный базис остается допустимым? В отличие от строки 2 строка 1 вычиталась в процессе выполнения симплексного алгоритма из других строк. Заметим, что появление δ в правой части каждого уравнения сопровождается появлением x_5 в левой части этого же уравнения с тем же коэффициентом. Следовательно, на последней итерации будем иметь:

$$\begin{aligned}
 x_0 + x_1 &+ 4x_4 + \frac{1}{2}x_5 + \frac{3}{5}x_7 = 58 + \frac{1}{2}\delta \quad (\text{строка } 0), \\
 x_1 + x_2 &- 2x_4 + x_5 - \frac{1}{5}x_7 = 4 + \delta \quad (\text{строка } 1), \\
 2x_1 &+ 4x_4 - \frac{5}{2}x_5 + x_6 + \frac{2}{5}x_7 = 82 - \frac{5}{2}\delta \quad (\text{строка } 2), \\
 x_3 + x_4 &- \frac{1}{2}x_5 + \frac{1}{5}x_7 = 6 - \frac{1}{2}\delta \quad (\text{строка } 3).
 \end{aligned} \tag{F1}$$

Чтобы базисные переменные x_2 , x_6 , x_3 были неотрицательными, правые части строк (1), (2), (3) также должны быть неотрицательными. Отсюда следует, что прежнее оптимальное решение остается допустимым, если $-4 \leq \delta \leq 12$. При $\delta < -4$ отрицательным становится x_2 , при $\delta > 12$ - x_3 .

Пусть $\delta = 1$, что соответствует увеличению ресурса, описываемого строкой 2 на единицу. Из строки 0 системы (F1) видно, что значение целевой функции возрастает при этом на $\frac{1}{2}$, причем прежний базис остается допустимым. То есть при увеличении данного ресурса на единицу дополнительная прибыль в оптимальном варианте составляет $\frac{1}{2}$.

Рассмотренный анализ можно обобщить на случай варьирования одновременно нескольких констант в правых частях ограничений. Произведем одновременно следующие замены:

$$\begin{aligned} 20 &\rightarrow 20 + \delta_1 && \text{в строке 1,} \\ 100 &\rightarrow 100 + \delta_2 && \text{в строке 2,} \\ 80 &\rightarrow 80 + \delta_3 && \text{в строке 3.} \end{aligned}$$

Если обратить в ноль все небазисные переменные, система уравнений (F) на последней симплексной итерации примет вид:

$$\begin{aligned} x_0 &= 58 + \frac{1}{2}\delta_1 && + \frac{3}{5}\delta_3, \\ x_2 &= 4 + \delta_1 && - \frac{1}{5}\delta_3, \\ x_6 &= 82 - \frac{5}{2}\delta_1 + \delta_2 + \frac{2}{5}\delta_3, \\ x_3 &= 6 - \frac{1}{2}\delta_1 && + \frac{1}{5}\delta_3. \end{aligned}$$

Коэффициенты δ_i ($i=1,2,3$) совпадают с коэффициентами при соответствующих остаточных переменных в (F). Базис остается допустимым, если x_2 , x_6 и x_3 неотрицательны. Следовательно, δ_1 , δ_2 и δ_3 должны удовлетворять соответствующей системе неравенств.

3.6.4. Двойственность

Понятие двойственности является очень важным и полезным, так как позволяет связать в единую систему различные методы анализа на чувствительность. Рассмотрим две следующие задачи линейного программирования:

$$\begin{aligned} &\sum_{j=1}^n c_j x_j \Rightarrow \max, \\ \text{а)} &\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = \overline{1, m}), \\ &x_j \geq 0 \quad (j = \overline{1, n}); \\ &\sum_{i=1}^m b_i y_i \Rightarrow \min, \\ \text{б)} &\sum_{i=1}^m a_{ij} y_i \geq c_j \quad (j = \overline{1, n}), \\ &y_i \geq 0 \quad (i = \overline{1, m}). \end{aligned}$$

Назовем первую задачу *исходной*, а вторую – *двойственной* (по отношению к первой). В качестве примера рассмотрим задачу распределения ресурсов:

$$\begin{aligned} 3x_1 + 4x_2 + 7x_3 + 9x_4 &\Rightarrow \max, \\ 2x_1 + 2x_2 + 2x_3 + 2x_4 &\leq 20, \\ 5x_1 + 3x_2 + x_3 + x_4 &\leq 100, \\ 5x_1 + 5x_2 + 10x_3 + 20x_4 &\leq 80, \\ x_j &\geq 0 \quad (j=1,2,3,4). \end{aligned}$$

Двойственная задача имеет вид:

$$\begin{aligned} 20y_1 + 100y_2 + 80y_3 &\Rightarrow \min, \\ 2y_1 + 5y_2 + 5y_3 &\geq 3, \\ 2y_1 + 3y_2 + 5y_3 &\geq 4, \\ 2y_1 + y_2 + 10y_3 &\geq 7, \\ 2y_1 + y_2 + 20y_3 &\geq 9, \\ y_1 \geq 0, y_2 \geq 0, y_3 &\geq 0. \end{aligned}$$

Грубо говоря, двойственная задача – это на 90 градусов повернутая исходная задача. Действительно:

- 1) j -й столбец коэффициентов в ограничениях исходной модели совпадает с j -й строкой коэффициентов в ограничениях двойственной задачи;
- 2) строка из коэффициентов целевой функции совпадает со столбцом из констант в правых частях ограничений двойственной модели;
- 3) столбец из констант, стоящих в правых частях ограничений исходной модели, совпадает со строкой из коэффициентов целевой функции двойственной модели;
- 4) направление знаков неравенств в исходной модели противоположно направлению знаков неравенств в двойственной модели, требование максимизации в исходной задаче заменено в двойственной модели требованием минимизации.

Имеют место две важные теоремы. Теорема 1 – **теорема двойственности**: если исходная и двойственная ей задачи имеют допустимые решения, то:

- 1) существует оптимальное решение x_j^* ($j = \overline{1, n}$) исходной задачи;
- 2) существует оптимальное решение y_i^* ($i = \overline{1, m}$) двойственной задачи;

3) имеет место соотношение: $\sum_{j=1}^n c_j x_j^* = \sum_{i=1}^m b_i y_i^*$.

Если исходная задача имеет неограниченное оптимальное решение, соответствующая ей двойственная задача вообще не имеет допустимых решений.

Теорема 2 – *теорема о дополнительной нежесткости*: пусть $x_j^* (j = \overline{1, n})$ – решение исходной задачи, а $y_i^* (i = \overline{1, m})$ – решение соответствующей двойственной задачи. Оба решения являются оптимальными тогда и только тогда, когда:

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

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

Отсюда вытекает, что всякий раз, когда модель содержит ограничение, принимающее после подстановки оптимального решения вид строгого неравенства, соответствующая переменная двойственной задачи принимает нулевое значение.

Для решения двойственной задачи можно использовать рассмотренный выше симплексный алгоритм. Но если решена исходная задача, то решение двойственной задачи является избыточным, так как справедливо:

а) коэффициенты при остаточных переменных в строке 0 на последней симплекс-итерации при решении исходной задачи совпадают с оптимальными значениями переменных двойственной задачи;

б) коэффициент при x_j в строке 0 на последней симплекс-итерации представляет собой разность между левой и правой частями j -го ограничения двойственной задачи, соответствующего оптимальному решению последней.

В качестве примера опять рассмотрим задачу распределения ресурсов. Из (F) видно, что оптимальные значения переменных двойственной задачи следующие:

$$y_1^* = \frac{1}{2}, y_2^* = 0, y_3^* = \frac{3}{5}.$$

При этом значение целевой функции двойственной задачи совпадает с значением целевой функции исходной задачи: $20 \cdot \frac{1}{2} + 100 \cdot 0 + 80 \cdot \frac{3}{5} = 58$.

Наконец, вычислим разность между левыми и правыми частями ограничений двойственной задачи. Например, для второго и третьего ограничений находим: $2 \cdot \frac{1}{2} + 5 \cdot 0 + 5 \cdot \frac{3}{5} = 4$, $4 - 4 = 0$, то есть, получаем соответственно коэффициенты при x_2 и x_3 в строке 0 системы (F).

3.6.5. Использование двойственности для анализа на чувствительность

Рассмотрим изменение констант в правых частях ограничений. Из проведенного ранее анализа на чувствительность при изменении констант в правых частях ограничений, видно, что коэффициент при остаточной переменной в строке 0 системы (F), соответствующей оптимальному решению, определяет увеличение прибыли при изменении соответствующего ресурса на единицу. С другой стороны, было показано, что значение данного коэффициента совпадает с оптимальным значением соответствующей переменной двойственной задачи. Отсюда следует вывод о том, что оптимальное значение каждой переменной двойственной задачи определяет положительное или отрицательное приращение целевой функции за счет единичного приращения (положительного или отрицательного) значения константы в правой части соответствующего ограничения при условии, что рассматриваемый базис остается допустимым. Это согласуется с теоремой двойственности, согласно которой:

$$x_0^* = \sum_{i=1}^m b_i y_i^* .$$

Оптимальные значения переменных двойственной задачи часто называют *скрытыми доходами*. Если константы в правых частях ограничений задают объемы имеющихся ресурсов, то скрытые доходы определяют вклад в прибыль, полученный за счет единицы каждого из ресурсов.

В рассмотренном ранее примере распределения ресурсов значение $\frac{1}{2}$ есть скрытый доход, соответствующий первому ограничению (ресурс – человеко-недели), значение 0 – скрытый доход, соответствующий второму ограничению (ресурс – объем материала Y), а значение $\frac{3}{5}$ – скрытый доход, соответствующий третьему ограничению (ресурс – материал Z). Таким образом, увеличив первый из указанных ресурсов на одну человеко-неделю, мы получаем дополнительную прибыль, равную $\frac{1}{2}$, каждый дополнительный килограмм материала Z увеличивает прибыль на $\frac{3}{5}$. Увеличение же объема материала Y не приводит к увеличению прибыли. Это обусловлено тем, что запас материала Y превышает потребности в нем, так как остаточная переменная x_6 входит в оптимальный базис.

Рассмотрим еще один вид анализа на чувствительность, существенную помощь в выполнении которого может оказать двойственность. Пусть рассматривается вопрос о том, вводить или не вводить в модель новую управляемую переменную. Естественно связать данный вопрос с вопросом определения оптимальности прежнего решения. Если прежнее решение остается оп-

тимальным и при добавлении новой переменной, то нет никакого смысла включать ее в модель.

Наиболее простой способ получить ответ об оптимальности прежнего решения заключается в том, чтобы проверить, удовлетворяются ли ограничения, соответствующие двойственной задаче. Если эти ограничения не выполняются, то следует ввести в рассмотрение новую переменную.

Пусть в задаче распределения ресурсов вводится дополнительная переменная x_8 , причем дополнения к строкам имеют вид:

$$\begin{aligned} &+2x_8 \text{ (строка 1),} \\ &+\frac{3}{5}x_8 \text{ (строка 2),} \\ &+15x_8 \text{ (строка 3).} \end{aligned}$$

Пусть при переменной x_8 в целевой функции стоит коэффициент c_8 . При каком значении c_8 целесообразно ввести в базис x_8 ? Соответствующее соотношение двойственной задачи имеет вид: $2y_1 + \frac{3}{5}y_2 + 15y_3 \geq c_8$. Подставив сюда оптимальные значения переменных двойственной задачи, получим: $2 \cdot \frac{1}{2} + \frac{3}{5} \cdot 0 + 15 \cdot \frac{3}{5} \geq c_8$, или $10 \geq c_8$.

Следовательно, при $c_8 > 10$ переменная x_8 войдет в оптимальный базис. Поэтому x_8 следует ввести в модель.

3.7. Примеры задач

Задача 3.1

Деревообрабатывающая фирма столкнулась с проблемой наиболее рационального использования древесины в одном из лесных массивов. Данная фирма может производить пиломатериалы, а так же фанеру.

Чтобы получить $2,5\text{ м}^3$ пиломатериалов, необходимо израсходовать $2,5\text{ м}^3$ еловых и $7,5\text{ м}^3$ пихтовых лесоматериалов. Для приготовления 100 м^2 фанеры требуется 5 м^3 еловых и 10 м^3 пихтовых лесоматериалов. Лесной массив содержит 100 м^3 еловых и 450 м^3 пихтовых лесоматериалов.

Для удовлетворения самого необходимого спроса в течение планового периода необходимо произвести, по крайней мере, 20 м^3 пиломатериалов и 2000 м^2 фанеры. Доход с 1 м^3 пиломатериалов составляет 20 рублей, а со 100 м^2 фанеры – 50 рублей.

Требуется построить линейную оптимизационную модель и найти оптимальное решение.

Решение

1. Введем управляемые переменные:

x_1 – количество (в м^3) производимых пиломатериалов;

x_2 – количество (в десятках м^2) изготавливаемой фанеры.

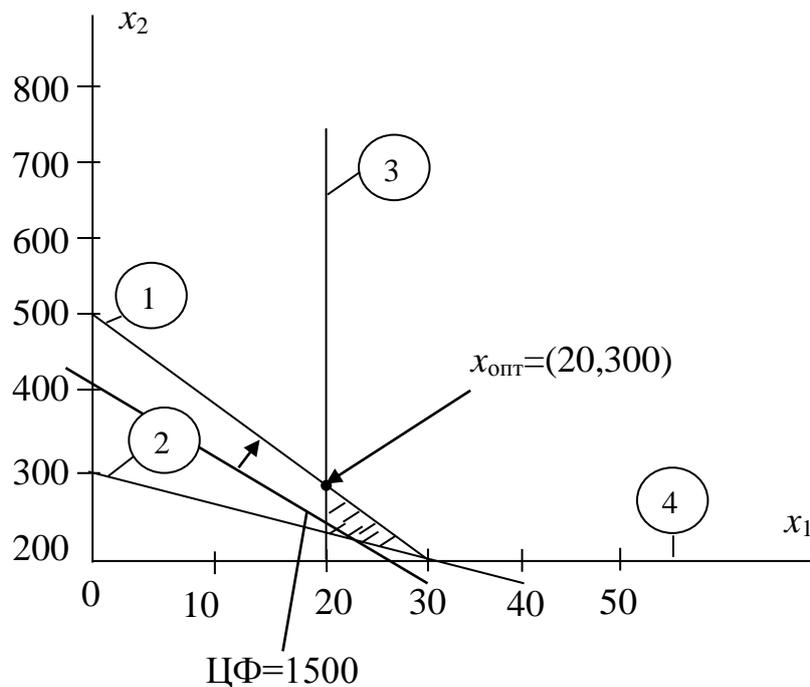
2. Математическая модель имеет вид:

$$\begin{aligned} 20x_1 + 5x_2 &\Rightarrow \max, \\ 2x_1 + 0,2x_2 &\leq 100, \\ 5x_1 + 1,5x_2 &\leq 450, \\ x_1 &\geq 20, \quad x_2 \geq 200. \end{aligned}$$

3. Для построения области допустимых решений построим уравнения прямых:

$$\begin{aligned} 2x_1 + 0,2x_2 &= 100, && \text{(линия 1)} \\ 5x_1 + 1,5x_2 &= 450, && \text{(линия 2)} \\ x_1 &= 20, && \text{(линия 3)} \\ x_2 &= 200. && \text{(линия 4)} \end{aligned}$$

Для удобства построения совместим ось абсцисс с прямой $x_2=200$ (рис. 14). Линию равного уровня целевой функции удобно построить для значения 1500. Двигая эту линию в сторону увеличения, получим точку $x_{\text{опт}}=(20,300)$. Оптимальное значение целевой функции: $20 \times 20 + 5 \times 300 = 1900$.



Задача 3.2

Рис. 14. Графическое решение задачи 3.1

Руководство нефтеперерабатывающей фирмы пытается оптимально распределить имеющиеся в ее распоряжении два сорта нефти (А и В) по двум возможным технологическим процессам составления смесей. Технологический процесс 1 характеризуется следующими показателями: из трех единиц объема нефти А и четырех единиц объема нефти В получаются пять единиц объема бензина Y и две единицы объема бензина Z. Технологический процесс 2 характеризуется другими показателями: из четырех единиц объема нефти А

и семи единиц объема нефти В получаются три единицы бензина Y и восемь единиц бензина Z.

Максимальное потребляемое количество нефти А равно 100 единиц объема, а нефти В – 150 единиц объема. По условиям поставок требуется произвести не менее 200 единиц объема бензина Y и 75 единиц объема бензина Z. Доходы с единицы объема продукции, получаемой с помощью технологических процессов 1 и 2, составляют p_1 и p_2 соответственно.

Требуется построить модель линейного программирования.

Решение

1. Введем управляемые переменные:

x_1 – объем продукции, выпускаемой технологическим процессом 1;

x_2 – объем продукции, выпускаемой технологическим процессом 2.

2. Математическая модель:

$$p_1x_1 + p_2x_2 \Rightarrow \max,$$

$$\frac{5}{7}x_1 + \frac{3}{11}x_2 \geq 200,$$

$$\frac{2}{7}x_1 + \frac{8}{11}x_2 \geq 75,$$

$$\frac{3}{7}x_1 + \frac{4}{11}x_2 \leq 100,$$

$$\frac{4}{7}x_1 + \frac{7}{11}x_2 \leq 150,$$

$$x_{1,2} \geq 0.$$

Задача 3.3

На одном из предприятий предстоит решить, какое количество чистой стали, и какое количество металлолома следует использовать для приготовления литья из сплава для одного из заказчиков. Пусть производственные затраты в расчете на одну тонну чистой стали равны 30 рублей, а затраты в расчете на одну тонну металлолома – 50 рублей (так как использование металлолома сопряжено с предварительной очисткой). Заказ предусматривает не менее 5 тонн литья, при этом заказчик готов купить и большее количество литья.

Известно, что запасы чистой стали ограничены и не превышают четырех тонн, а запасы металлолома не превышают шести тонн. Отношение веса металлолома к весу чистой стали в процессе получения сплава не должно превышать 7:8. Производственно-технологические условия таковы, что на процессы плавки и литья не может быть отведено более 18 часов, при этом на 1 тонну стали уходит 3 часа, а на 1 тонну металлолома – 2 часа производственного времени. Требуется: а) построить линейную оптимизационную модель; б) графически найти оптимальное решение.

Решение

1. Введем управляемые переменные:

x_1 – количество стали (тонн);

x_2 – количество металлолома (тонн).

2. Запишем целевую функцию и ограничения:

$$40x_1 + 60x_2 \rightarrow \min,$$

$$x_1 + x_2 \geq 10,$$

$$x_1 \leq 5,$$

$$x_2 \leq 8,$$

$$9x_1 - 10x_2 \geq 0,$$

$$4x_1 + 2x_2 \leq 32,$$

$$x_{1,2} \geq 0.$$

3. Графическое решение модели приведено на рис.15. Для построения области допустимых решений используются уравнения прямых:

$$x_1 + x_2 = 10, \quad (\text{линия 1})$$

$$x_1 = 5, \quad (\text{линия 2})$$

$$x_2 = 8, \quad (\text{линия 3})$$

$$9x_1 - 10x_2 = 0, \quad (\text{линия 4})$$

$$4x_1 + 2x_2 = 32, \quad (\text{линия 5})$$

$$x_1 = 0, \quad (\text{линия 6})$$

$$x_2 = 0, \quad (\text{линия 7}).$$

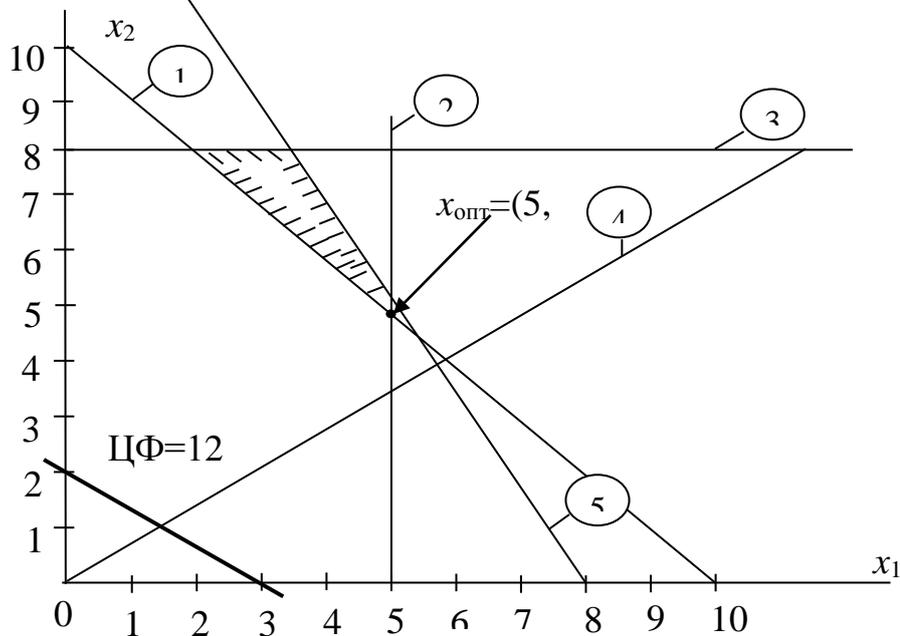


Рис. 15. Графическое решение задачи 3.3

На рис.15 построена линия равного уровня целевой функции, равная 120. Оптимальное (минимальное) значение целевой функции достигается в точке $(x_1=5, x_2=5)$: $40 \times 5 + 60 \times 5 = 500$.

Задача 3.4

Фирма производит холодильники, газовые плиты и кухонные раковины. В настоящем году ожидается спрос, приведенный на рис.16. Фирма разрабатывает годовой план, который был бы в состоянии удовлетворить спрос. Удовлетворение квартального спроса может быть произведено как за счет производства в данном квартале, так и за счет ранее сделанных запасов. В начале первого и в конце четвертого кварталов запасы отсутствуют. Известно, что хранение каждой единицы продукции на складе в течение квартала обходится в пять единиц денег.

	Кварталы			
	1	2	3	4
Холодильники	2000	1500	3000	1000
Газовые плиты	1500	1500	1000	1500
Кухонные раковины	1000	3000	1500	3000

Рис. 16. Спрос на продукцию в задаче 3.4

В течение квартала фирма может израсходовать не более 8000 часов рабочего времени. На изготовление холодильника требуется 0,5 часа, газовой плиты – 2 часа, раковины – 1,5 часа. В четвертом квартале холодильники изготавливаться не могут, так как фирма планирует в это время провести модернизацию оборудования.

Требуется построить линейную оптимизационную модель, которая может помочь при нахождении хорошего плана.

Решение

1. Вводим управляемые переменные:

x_{it} – объем производства в t -м квартале ($t=1,2,3,4$) холодильников ($i=1$), плит ($i=2$), раковин ($i=3$), причем x_{14} тождественно равен нулю;

e_{it} – запас на конец t -го квартала ($t=1,2,3$) холодильников ($i=1$), плит ($i=2$), раковин ($i=3$).

2. Записываем линейную динамическую модель:

$$\begin{aligned}
 &5e_{11} + 5e_{12} + 5e_{13} + 5e_{21} + 5e_{22} + 5e_{23} + 5e_{31} + 5e_{32} + 5e_{33} \Rightarrow \min, \\
 &\quad 0,5x_{11} + 2x_{21} + 1,5x_{31} \leq 8000, \\
 &x_{11} \quad \quad \quad -e_{11} \geq 2000, \\
 &\quad x_{21} \quad \quad \quad -e_{21} \geq 1500, \\
 &\quad \quad x_{31} \quad \quad -e_{31} \geq 1000, \\
 &\quad \quad \quad 0,5x_{12} + 2x_{22} + 1,5x_{32} \leq 8000,
 \end{aligned}$$

$$\begin{array}{rcccccl}
x_{12} & & +e_{11} & & -e_{12} & \geq 1500, \\
x_{22} & & +e_{21} & & -e_{22} & \geq 1500, \\
& x_{32} & +e_{31} & & -e_{32} & \geq 3000, \\
0,5x_{13} + 2x_{23} + 1,5x_{33} & & & & & \leq 8000, \\
x_{13} & & +e_{12} & & -e_{13} & \geq 3000, \\
& x_{23} & & +e_{22} & & -e_{23} \geq 1000, \\
& & x_{33} & +e_{32} & & -e_{33} \geq 1500, \\
2x_{24} + 1,5x_{34} & & & & & \leq 8000, \\
& & & +e_{13} & & \geq 1000, \\
& x_{24} & & +e_{23} & & \geq 1500, \\
x_{34} & & & +e_{33} & & \geq 3000, \\
x_{it}, e_{it} \geq 0 \quad (i=1,2,3, \quad t=1,2,3,4).
\end{array}$$

Задача 3.5

Известны минимальные потребности районного отделения милиции в количестве милиционеров в различное время суток (рис.17). При этом нужно иметь в виду, что период 1 следует сразу же за периодом 6. Каждый милиционер работает 8 часов без перерыва. Начальник отделения хочет составить служебное расписание на сутки таким образом, чтобы обойтись минимальным числом милиционеров, но не нарушая сформулированных выше требований.

Время суток, часы	Порядковый номер периода	Минимальное число милиционеров, требуемых в указанный период
2-6	1	20
6-10	2	50
10-14	3	80
14-18	4	100
18-22	5	40
22-2	6	30

Рис.17. Потребности в милиционерах в задаче 3.5

Требуется построить модель линейного программирования для составления такого расписания.

Решение

1. Вводим управляемые переменные: $x_j (j=\overline{1,6})$ - число милиционеров, приступающих к работе в начале j -го периода.

2. Запишем целевую функцию и ограничения:

$$\begin{array}{rcl}
x_1 + x_2 + x_3 + x_4 + x_5 + x_6 & \Rightarrow & \min \\
x_1 & + & x_6 \geq 20, \\
x_1 + x_2 & & \geq 50, \\
x_2 + x_3 & & \geq 80,
\end{array}$$

$$\begin{aligned}
 x_3 + x_4 &\geq 100, \\
 x_4 + x_5 &\geq 40, \\
 x_5 + x_6 &\geq 30, \\
 x_j &\geq 0 \quad (j=\overline{1,6}).
 \end{aligned}$$

4. СЕТЕВЫЕ ОПТИМИЗАЦИОННЫЕ МОДЕЛИ

4.1. Общие свойства сетевых моделей

Сетевой моделью называется математическая модель, структура которой может быть изображена в виде графа, называемого сетью. *Сеть* есть множество узлов (вершин) и множество дуг (ребер), соединяющих различные пары узлов. На каждой дуге задана определенная ориентация, поэтому сеть является ориентированной. Весьма общей сетевой оптимизационной моделью является модель:

$$\sum_{i=1}^p \sum_{j=1}^p c_{ij} x_{ij} \Rightarrow \min, \quad (4.1)$$

$$\sum_{j=1}^p x_{kj} - \sum_{i=1}^p x_{ik} = T_k, \quad k = \overline{1, p}, \quad (4.2)$$

$$x_{ij} \geq 0, \quad i = \overline{1, p}, \quad j = \overline{1, p}. \quad (4.3)$$

При этом ограничения (4.2) имеют ту важную особенность, что все коэффициенты при управляемых переменных могут принимать одно из двух ненулевых значений, а именно: $+1$ и -1 . Каждая из переменных входит, как правило, в два ограничения. При этом один из коэффициентов при переменной равен $+1$, а другой -1 .

Рассмотрим сетевую интерпретацию элементов модели (4.1)–(4.3). Этой модели соответствует сеть с p узлами. В некоторых из этих узлов имеются излишки однородного для данной сети ресурса, обозначаемые как T_k ($T_k > 0$). В других узлах недостатки этого ресурса в объеме T_k ($T_k < 0$). Перемещение каждой единицы ресурса из узла i в узел j обходится в c_{ij} единиц затрат. Управляемая переменная x_{ij} показывает, сколько единиц ресурса будет перенесено в плановый период по дуге (i, j) . Таким образом, каждое из p ограничений (4.2) требует, чтобы разность между суммарным выходом ресурса из данного узла k и суммарным поступлением ресурса в этот узел равнялась избытку (дефициту) ресурса в данном узле.

В результате решения сетевой задачи находится оптимальное решение, которое показывает: по каким дугам необходимо переместить какие количе-

ства ресурсов, чтобы с минимальными затратами удовлетворить дефициты в некоторых узлах за счет избытков ресурса в других.

Иногда для приведения модели к виду (4.1)-(4.3) приходится объединять уравнения, изменять знаки коэффициентов и вводить дополнительное соотношение или рассматривать двойственную задачу. Однако после приведения модели к такому виду каждое уравнение (4.2) соответствует узлу сети, каждая переменная x_{ij} – величине потока по дуге (i, j) , а каждая величина c_{ij} равна затратам на доставку единицы ресурса из узла i в узел j по дуге (i, j) .

Преимущество сетевой модели по сравнению с соответствующей ей задачей линейного программирования заключается, во-первых, в большей эффективности алгоритма оптимизации на сетевой модели по сравнению со стандартным симплексным алгоритмом, а во-вторых – в наглядности описания структуры модели с помощью сети.

4.2. Модель назначений

4.2.1. Описание модели

Данная модель является простейшим типом сетевых моделей. Задачу о назначениях можно кратко сформулировать следующим образом. Имеется n работ, каждую из которых может выполнить любой из n исполнителей. Стоимость выполнения работы i исполнителем j равна c_{ij} . Нужно распределить исполнителей по работам (то есть назначить одного исполнителя на какую-то одну работу) таким образом, чтобы минимизировать общие затраты.

В качестве примера рассмотрим предприятие, выпускающее сложную электронную аппаратуру, которому необходимо изготовить несколько тысяч новых изделий, собираемых из отдельных блоков. Руководство предприятия приняло решение разместить заказы на изготовление n блоков и выбрало n предприятий–поставщиков. Каждый заказ настолько велик, что один поставщик не может выполнить более одного заказа. Каждому поставщику было предложено определить стоимость изготовления каждого из блоков, то есть c_{ij} . Располагая этой информацией необходимо распределить заказы по поставщикам.

Задача о назначениях описывается сетью, приведенной на рис.18.

Для записи соответствующей математической модели введем управляемые переменные x_{ij} :

$$x_{ij} = \begin{cases} 1, & \text{если заказ } i \text{ выполняется поставщиком } j, \\ 0, & \text{в остальных случаях,} \end{cases}$$

где $i = \overline{1, n}$, $j = \overline{1, n}$.

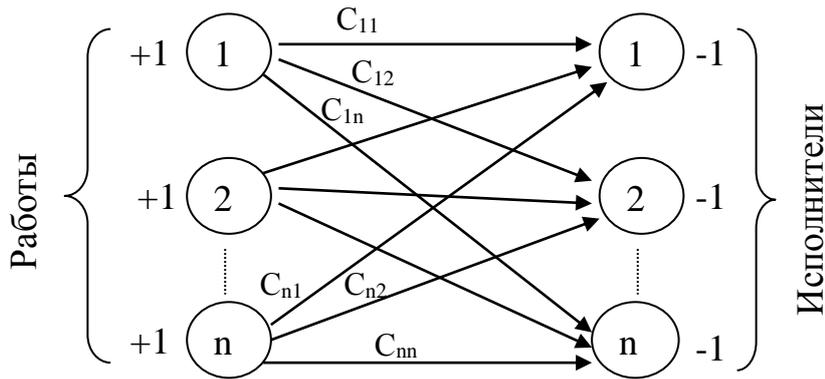


Рис. 18. Сеть задачи о назначениях

Математическая модель имеет вид:

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

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

$$\sum_{i=1}^n x_{ij} = 1, \quad j = \overline{1, n}, \quad x_{ij} = 0, 1 \quad (i = \overline{1, n}, j = \overline{1, n}).$$

4.2.2. Алгоритм решения задачи о назначениях

Изложение данного алгоритма начнем с записи матрицы, приведенной на рис. 19.

C_{11}	C_{12}	...	C_{1n}
C_{21}	C_{22}	...	C_{2n}
		...	
C_{n1}	C_{n2}	...	C_{nn}

Рис.19. Матрица задачи о назначениях

Алгоритм состоит из следующих шагов.

Шаг 1. В каждой строке матрицы найти наименьший элемент и вычесть его из всех элементов данной строки. Затем то же самое сделать с каждым столбцом полученной матрицы. Правомочность данного шага обусловлена тем, что все исполнители находятся в равных условиях по отношению к i -той работе, когда мы из всех c_{ij} ($j = \overline{1, n}$) вычитаем одну и ту же константу s . Оптимальный набор управляемых переменных для преобразованной матрицы остается прежний, хотя оптимальное значение целевой функции и уменьшится на s . То же относится и к столбцам.

Шаг 2. Зачеркиваем все нули преобразованной матрицы минимальным числом прямых, проходящим по строкам и по столбцам. Если это число меньше n , то переход к шагу 3, если оно равно n , то оптимальное решение получено. При этом каждой оптимальной управляемой переменной $x_{ij}^* = 1$ соответствует *независимый ноль* – такой ноль, в строке и столбце которого нет других независимых нулей. Оптимальность данного решения для преобразованной матрицы очевидно, так как целевая функция принимает при этом значение 0. При этом оптимальное значение целевой функции исходной задачи есть сумма соответствующих c_{ij} .

Шаг 3. Среди не зачеркнутых элементов матрицы выбираем наименьший и вычитаем его из всех не зачеркнутых элементов и прибавляем его ко всем элементам, зачеркнутым дважды. Правомочность данного шага обусловлена тем, что он эквивалентен вычитанию указанного числа из каждой не зачеркнутой строки и прибавлению его к каждому зачеркнутому столбцу.

Переход на шаг 2.

Пример. Требуется найти оптимальное решение задачи о назначениях, исходная матрица которой приведена на рис. 20.

3	13	18	5
12	21	23	13
13	27	29	18
11	28	30	15

Рис. 20. Исходная матрица задачи о назначениях

Шаг 1. Выполним вычитание минимальных элементов по строкам, а затем по столбцам (рис. 21).

0	10	15	2
0	9	11	1
0	14	16	5
0	17	19	4

,

0	1	4	1
0	0	0	0
0	5	5	4
0	8	8	3

Рис. 21. Выполнение шага 1

Итерация 1

Шаг 2. Выполним зачеркивание всех нулей минимальным числом прямых (рис.22). Так как таких прямых всего 2, то оптимальное решение не получено.

0	1	4	1
0	0	0	0
0	5	5	4
0	8	8	3

Рис. 22. Выполнение шага 2 итерации 1

Шаг 3. Среди не зачеркнутых элементов матрицы выбираем наименьший (1) и вычитаем его из всех не зачеркнутых элементов, а затем прибавляем его ко всем элементам, зачеркнутым дважды (рис. 23).

0	0	3	0
1	0	0	0
0	4	4	3
0	7	7	2

Рис. 23. Результат выполнения шага 3 итерации 1

Итерация 2

Шаг 2. Выполним зачеркивание всех нулей минимальным числом прямых (рис. 24). Так как таких прямых всего 3, то оптимальное решение не получено.

0	0	3	0
1	0	0	0
0	4	4	3
0	7	7	2

Рис. 24. Выполнение шага 2 итерации 2

Шаг 3. Среди не зачеркнутых элементов матрицы выбираем наименьший (2) и вычитаем его из всех не зачеркнутых элементов, а затем прибавляем его ко всем элементам, зачеркнутым дважды (рис. 25).

2	0	3	0
3	0	0	0
0	2	2	1
0	5	5	0

Рис. 25. Результат выполнения шага 3 итерации 2

Итерация 3

Шаг 2. Выполним зачеркивание всех нулей минимальным числом прямых (рис. 26). Так как таких прямых 4, то, следовательно, можно найти 4 не-

зависимых нуля. Отметим эти нули символом «*». Независимым нулям соответствует оптимальное решение: $x_{12}=1$, $x_{23}=1$, $x_{31}=1$, $x_{44}=1$.

$$x_0^* = c_{12} + c_{23} + c_{31} + c_{44} = 13 + 23 + 13 + 15 = 64.$$

2	0*	3	0
3	0	0*	0
0*	2	2	1
0	5	5	0*

Рис. 26. Выполнение шага 2 итерации 3

4.3. Модель выбора кратчайшего пути

4.3.1. Общее описание модели

Несмотря на то, что это одна из самых простых сетевых моделей, ее практическое значение весьма велико. Кратко сущность модели состоит в следующем. Пусть задана сеть, каждой дуге которой соответствует некоторое расстояние c_{ij} . Требуется найти кратчайший путь в заданный узел, называемый **конечным узлом** или **стоком** из какого-то другого узла сети, который называется **начальным узлом** или **исток**. Иногда требуется найти кратчайшие пути до стока из всех остальных узлов сети.

Часто c_{ij} измеряется не в единицах длины. Например, c_{ij} может представлять собой стоимость переезда из узла i в узел j . В этом случае задача заключается в отыскании ориентированной цепи минимальной стоимости. Иногда c_{ij} есть время переезда из одного узла в другой. При этом требуется отыскать путь наименьшей продолжительности.

На рис. 27 приведен пример сети для задачи выбора кратчайшего пути. В данной сети много циклов. При этом **циклом** называется замкнутый путь, по которому с учетом ориентации дуг действительно может быть перемещен распределяемый на сети ресурс. Если общая длина пути вдоль какого-то цикла отрицательна, то многократно обойдя этот цикл, можно обеспечить бесконечное отрицательное значение целевой функции. Поэтому, если в сети есть циклы, общая длина любого цикла должна быть неотрицательна. Обычно это выполняется, так как $c_{ij} \geq 0$.

Сетевой модели выбора кратчайшего пути соответствует следующая модель линейного программирования:

$$\sum_{(i,j) \in \text{сети}} c_{ij} x_{ij} \Rightarrow \min, \quad (4.4)$$

$$\sum_{(k,j) \in \text{сети}} x_{kj} - \sum_{(i,k) \in \text{сети}} x_{ik} = \begin{cases} 1, & k = s \text{ (исток)}, \\ -1, & k = r \text{ (сток)}, \\ 0, & \text{для всех остальных } k, \end{cases} \quad (4.5)$$

$$x_{ij} \geq 0 \text{ для всех } (i, j) \in \text{сети}. \quad (4.6)$$

В (4.5) содержится по одному уравнению для каждого узла и по одной переменной для каждой дуги.

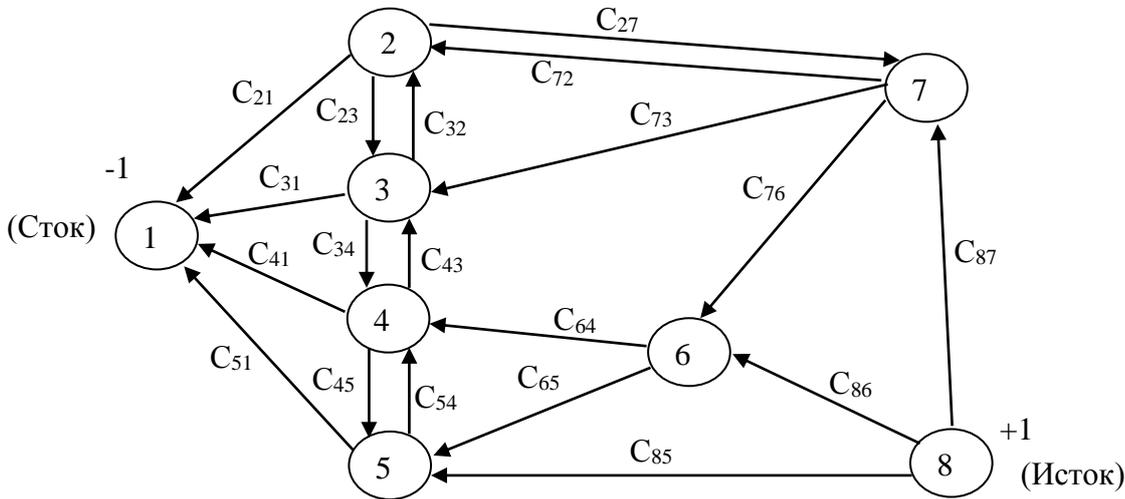


Рис. 27. Пример сети для поиска кратчайшего пути

Например, для сети на рис.27 первые два уравнения в (4.5) имеют вид:

$$x_{85} + x_{86} + x_{87} = 1 \quad (\text{узел 8 - исток}),$$

$$x_{72} + x_{73} + x_{76} - x_{27} - x_{87} = 0 \quad (\text{узел 7}).$$

4.3.2. Задача замены оборудования

Важным применением модели выбора кратчайшего пути является задача замены оборудования. Пусть некоторое предприятие разрабатывает план эксплуатации дорогостоящего оборудования на период $n-1$ лет. Предприятие может купить оборудование в начале года 1 и эксплуатировать его до начала

года $j \leq n$. Если $j < n$, то предприятие заменяет оборудование в начале года j и эксплуатирует новое до начала года $k \leq n$ и так далее. Величина затрат c_{ij} включает затраты на покупку оборудования в начале года i плюс затраты на ремонт и обслуживание оборудования, купленного в начале года i и замененного в начале года j .

Сеть, описывающая эту задачу при $n=6$, показана на рис. 28. В этой сети оборудование «отправляется» из узла 1 в узел 6. Каждый промежуточный узел соответствует году, когда должна производиться замена. Следует отметить, что данная сеть является ациклической, то есть она не содержит циклов.

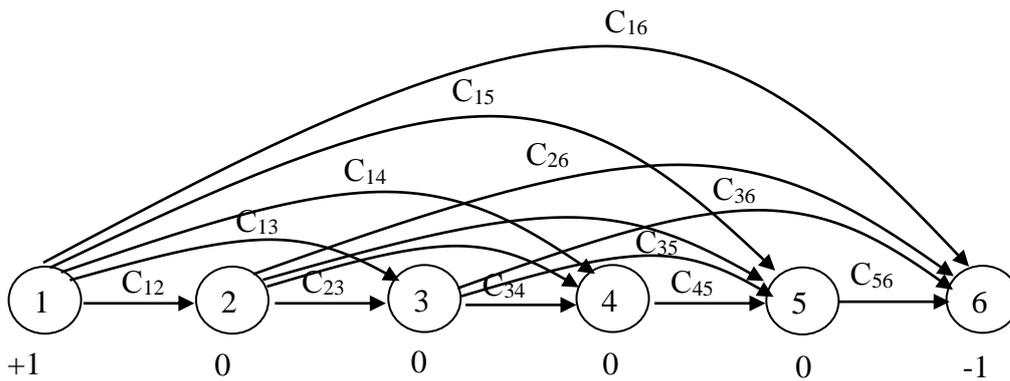


Рис. 28. Сеть для задачи замены оборудования

4.3.3. Оптимальное распределение капиталовложений

Некоторая фирма разрабатывает план капиталовложений на предстоящий год. Общая сумма средств, которой она располагает и которую нужно распределить по различным объектам, составляет B млн. рублей. Рассматриваются N возможных объектов капиталовложений. Минимальный объем вложений, необходимый для возведения объекта i , составляет p_i . При этом готовый объект будет давать прибыль v_{i0} . Но если в объект вложить дополнительно k млн. рублей, то объект будет давать прибыль v_{ik} , причем $v_{i,k+1} \geq v_{i,k}$. Необходимо так распределить капиталовложения по объектам, чтобы получить максимальную прибыль.

Пусть $B=6$, $N=3$, а минимальные объемы капиталовложений составляют соответственно $p_1=4$, $p_2=2$, $p_3=1$. Соответствующая сеть приведена на рис. 29. Она построена следующим образом. Для каждого объекта капиталовложений вводится столбец узлов. В обозначении узла (i, b) i есть номер объекта, b – сумма средств, которые можно вложить во все оставшиеся объекты, начиная с i . Каждая дуга, исходящая из узла (i, b) соответствует конкретному решению по объекту i .

Любой допустимый вариант распределения средств по объектам можно представить в виде ориентированного пути, который начинается в узле $(1,6)$ и

заканчивается в узле (4,0). Это означает, что весь объем средств должен быть полностью распределен между тремя объектами.

В этой задаче вес каждой дуги есть прибыль от принятия соответствующего решения. Поэтому необходимо отыскать в сети путь максимальной длины. Изменив знак целевой функции, задача о максимальном пути преобразуется в задачу о кратчайшем пути. Так как сеть ациклическая наличие отрицательных весов не приведет к зацикливанию.

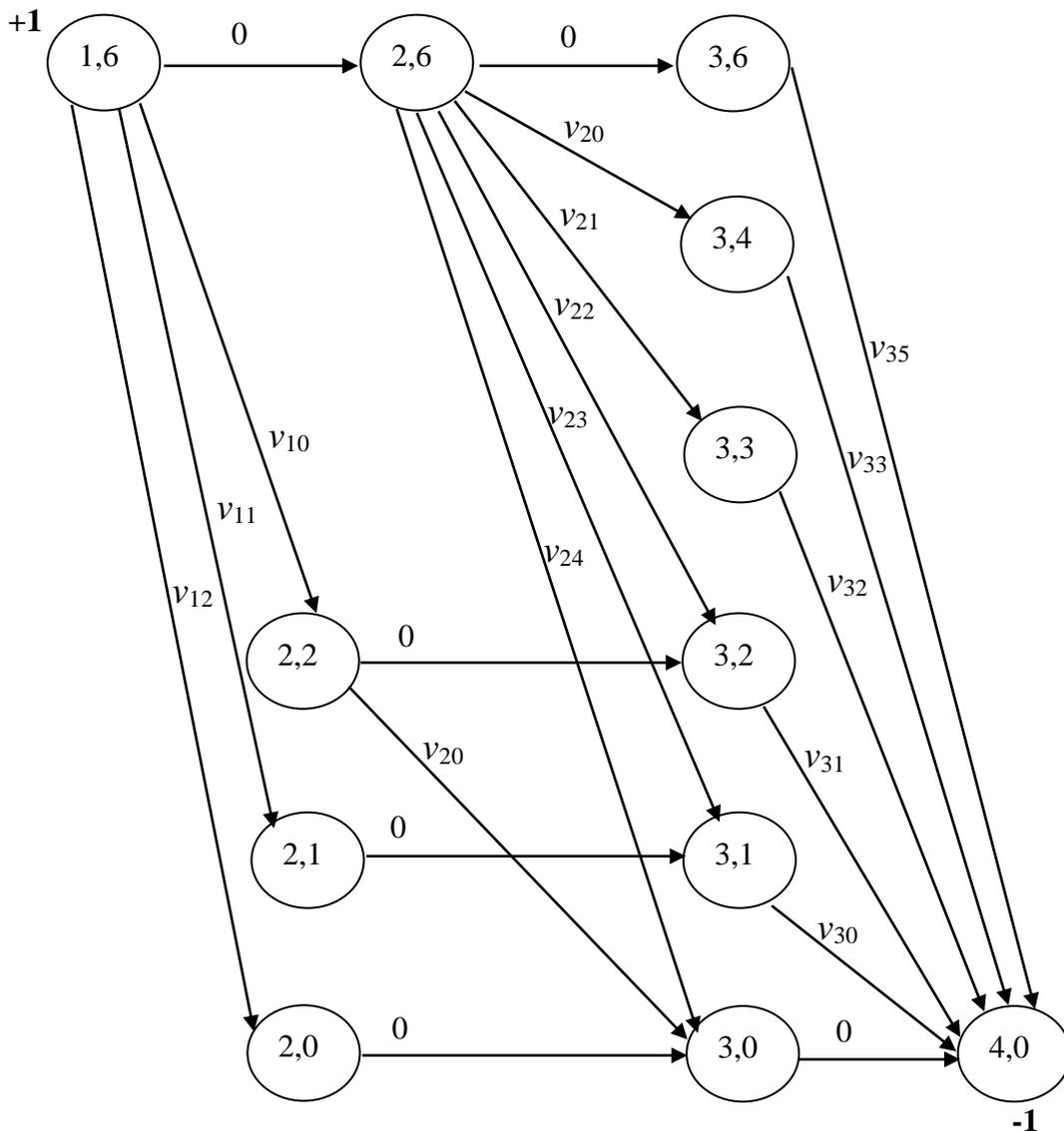


Рис. 29. Сеть для задачи распределения капиталовложений

Как и всякую задачу выбора кратчайшего пути данную задачу можно записать в виде задачи линейного программирования. Но кроме нее можно построить еще одну математическую модель, соответствующую данной сети.

Обозначим через x_i объем средств, вложенных в объект i , и определим неубывающую функцию прибыли от объекта i – $C_i(x_i)$:

$$C_i(x_i) = \begin{cases} 0, & x_i < p_i, \\ v_{ik}, & x_i = p_i + k, \quad k = 0, 1, 2, \dots \end{cases}$$

Тогда модель записывается в виде:

$$\sum_{i=1}^n C_i(x_i) \Rightarrow \max,$$

$$\sum_{i=1}^n x_i \leq B,$$

$$x_i = 0, 1, 2, \dots \text{ для любого } i.$$

В этом примере показано, как сетевую модель можно применить для решения задачи оптимизации нелинейной целевой функции в случае, когда на переменные наложено условие целочисленности и они подчиняются единственному ограничению.

4.3.4. Алгоритм нахождения кратчайшего пути в сети общего вида

В сети общего вида, в отличие от ациклической, могут быть циклы. Данный алгоритм основан на решении задачи, двойственной по отношению к задаче линейного программирования, описывающей данную сеть.

Ранее при формулировке исходной и двойственной задач нами была использована удобная простая каноническая форма. Запишем теперь исходную и двойственную задачи, воспользовавшись другими более сложными каноническими формами представления:

а) исходная задача

$$\sum_{i=1}^m b_i y_i \Rightarrow \max,$$

$$\sum_{i=1}^m a_{ij} y_i \leq c_j \quad (j = \overline{1, k}),$$

$$\sum_{i=1}^m a_{ij} y_i = c_j \quad (j = \overline{k+1, n}),$$

$$y_i \geq 0 \quad (i = \overline{1, h}),$$

б) двойственная задача

$$\begin{aligned} \sum_{j=1}^n c_j x_j &\Rightarrow \min, \\ \sum_{j=1}^m a_{ij} x_j &\geq b_i \quad (i = \overline{1, h}, h \leq m), \\ \sum_{j=1}^n a_{ij} x_j &= b_i \quad (i = \overline{h+1, m}), \\ x_j &\geq 0 \quad (j = \overline{1, k}, k \leq n); \end{aligned}$$

y_i не имеют ограничений в знаке при $i = \overline{h+1, m}$.

В этом случае также выполняются теоремы двойственности и дополнительной нежесткости. Частным случаем канонической формы исходной задачи является линейная модель (4.4)-(4.6), описывающая задачу поиска кратчайшего пути.

Двойственная по отношению к ней задача имеет вид:

$$-y_r + y_s \Rightarrow \max, \quad (4.7)$$

$$y_i - y_j \leq c_{ij} \quad \text{для всех } (i, j) \in \text{сети}, \quad (4.8)$$

где все y_i не ограничены в знаке.

Идея рассматриваемого алгоритма заключается в поиске оптимального решения двойственной задачи. Алгоритм состоит из следующих двух шагов.

Шаг 1. Принимается $y_r=0$, а все остальные $y_k=\infty$. Естественно, что данное пробное решение не удовлетворяет ограничениям (4.8).

Шаг 2. Если в сети остается хотя бы одна дуга (i, j) такая, что $y_i > c_{ij} + y_j$, то соответствующее значение y_i заменяется на $c_{ij} + y_j$. Иначе – оптимальное решение двойственной задачи найдено.

Следует отметить, что в процессе выполнения данного алгоритма находится кратчайший путь не только из истока до стока, но и из всех остальных вершин до стока. Найденное оптимальное решение двойственной задачи y_i^* ($i = \overline{1, p}$) есть длина кратчайшего пути из узла i до стока.

Пример. Требуется найти кратчайший путь для сети на рис.30, которая представляет собой сеть на рис.27 с численными значениями параметров c_{ij} .

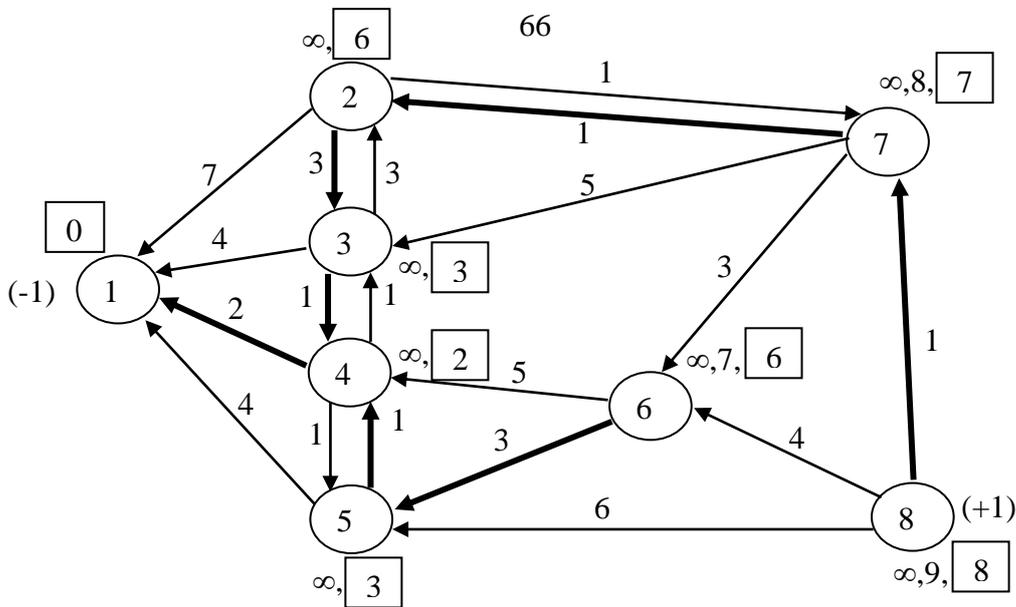


Рис. 30. Пример поиска кратчайшего пути в сети общего вида

Начнем вычисления с того, что проставим 0 у узла 1 и ∞ у всех остальных узлов (рис. 30). Далее можно выполнять вычисления в различном порядке, например, в таком:

$$y_4 = \infty > c_{41} + y_1 = 2 + 0, \text{ поэтому примем } y_4 = 2,$$

$$y_3 = \infty > c_{34} + y_4 = 1 + 2, \text{ поэтому примем } y_3 = 3,$$

$$y_6 = \infty > c_{64} + y_4 = 5 + 2, \text{ поэтому примем } y_6 = 7,$$

$$y_5 = \infty > c_{54} + y_4 = 1 + 2, \text{ поэтому примем } y_5 = 3,$$

$$y_6 = 7 > c_{65} + y_5 = 3 + 3, \text{ поэтому примем } y_6 = 6,$$

$$y_8 = \infty > c_{85} + y_5 = 6 + 3, \text{ поэтому примем } y_8 = 9,$$

$$y_2 = \infty > c_{23} + y_3 = 3 + 3, \text{ поэтому примем } y_2 = 6,$$

$$y_7 = \infty > c_{73} + y_3 = 5 + 3, \text{ поэтому примем } y_7 = 8,$$

$$y_7 = 8 > c_{72} + y_2 = 1 + 6, \text{ поэтому примем } y_7 = 7,$$

$$y_8 = 9 > c_{87} + y_7 = 1 + 7, \text{ поэтому примем } y_8 = 8.$$

При вычислении нового значения каждой величины y_i нужно обязательно проставлять найденное значение у узла i на рисунке. Оптимальные значения y_i^* выделены на рис.30 квадратами, а дуги, принадлежащие кратчайшим путям, выделены жирными линиями.

Докажем, что y_s^* действительно есть длина кратчайшего пути из узла s в узел r . По построению все ограничения двойственной задачи выполняются. Значение целевой функции двойственной задачи равно: $-y_r + y_s = 0 + y_s = y_s$. Это значение есть длина допустимого пути из s в r . Данный путь есть набор

дуг, который представляет собой допустимое решение исходной задачи. Так как допустимое решение исходной задачи имеет одинаковое значение целевой функции с допустимым решением двойственной задачи, то по теореме двойственности найденный путь является оптимальным.

4.3.5. Алгоритм нахождения кратчайшего пути в ациклической сети

Алгоритм отыскания кратчайшего пути можно упростить, если сеть является ациклической, так как в этом случае двойственные переменные y_i можно определять последовательно. Шаги алгоритма:

Шаг 1. Пронумеруем узлы сети от 1 до p таким образом, что если сеть содержит дугу (i, j) , то $i > j$. Для этого присвоим стоку номер 1. Зачеркнем этот узел и все входящие в него дуги и не будем их рассматривать в дальнейшем при нумерации. Возьмем любой другой узел, имеющий только входящие в него дуги, и припишем ему номер 2. Зачеркнем этот узел и все входящие в него дуги, и так далее пока все узлы не будут пронумерованы.

Шаг 2. Последовательно делаются следующие присвоения:

$$y_1 = 0 \text{ (сток),}$$

$$y_i = \min_{(i,j) \in \text{сети}} (c_{ij} + y_j) \text{ при } i = 2, 3, \dots, p.$$

Пример. Требуется найти кратчайший путь для сети на рис.31.

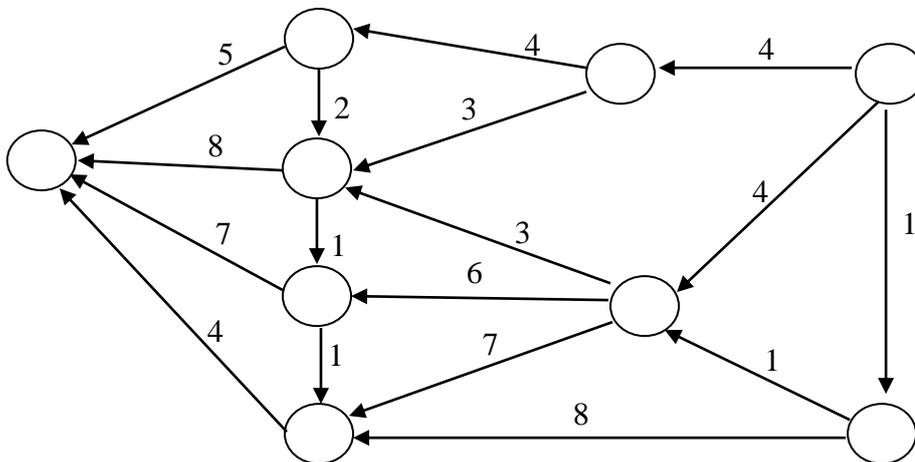


Рис. 31. Исходная ациклическая сеть

Шаг 1. Пронумеруем узлы сети (рис.32).

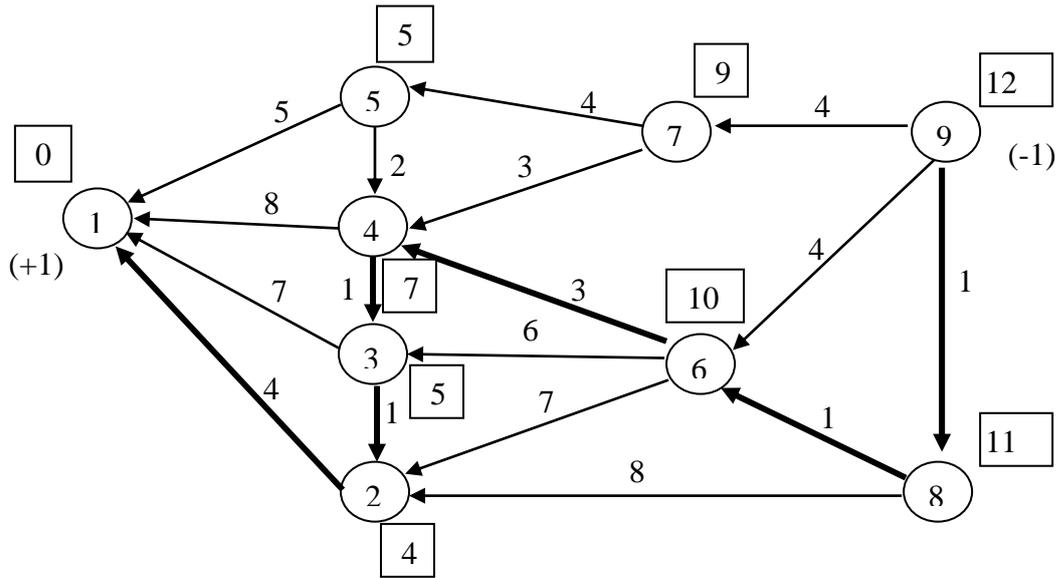


Рис. 32. Пример поиска кратчайшего пути в ациклической сети

Шаг 2. Запишем у каждой вершины i соответствующее значение y_i (рис. 32):

$$\begin{aligned}
 y_1 &= 0, \\
 y_2 &= \min(4+0) = 4, \\
 y_3 &= \min(7+0, 4+1) = 5, \\
 y_4 &= \min(8+0, 6+1) = 7, \\
 y_5 &= \min(5+0, 7+2) = 5, \\
 y_6 &= \min(7+3, 5+4) = 9, \\
 y_7 &= \min(4+8, 10+1) = 11, \\
 y_8 &= \min(11+1, 10+4, 9+4) = 12.
 \end{aligned}$$

Дуги, принадлежащие кратчайшим путям, выделены на рис.32 жирными линиями.

4.4. Транспортная задача

4.4.1. Модель классической транспортной задачи

Кратко сущность классической транспортной задачи состоит в следующем. Пусть имеются m различных поставщиков (предприятий или пунктов отправления), располагающих продукцией одного и того же типа, которую они могут отправить n потребителям. При этом предприятие i может отгрузить не более S_i единиц продукции, а потребителю j требуется не менее D_j единиц. Затраты на перевозку единицы груза из пункта отправления i в пункт

назначения j равны c_{ij} . Требуется так распределить потребителей по поставщикам, чтобы минимизировать общие транспортные затраты. Сеть, соответствующая данной задаче, приведена на рис.33.

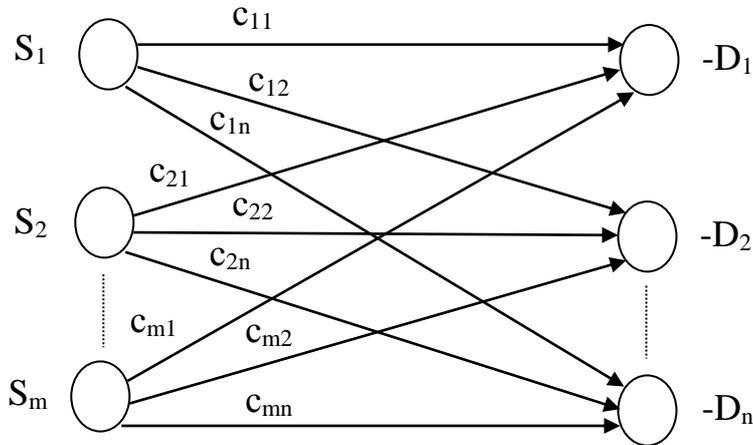


Рис. 33. Сеть классической транспортной задачи

Данной сети соответствует следующая задача линейного программирования:

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

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

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

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

Один из важнейших результатов, полученных в теории транспортных сетей, состоит в том, что если все S_i и D_j положительные целые числа, то среди всех оптимальных решений задачи (4.9)-(4.12) существует, по крайней мере, одно решение, в котором все значения x_{ij} являются целочисленными. Поэтому введение вместо (4.12) более сильного условия $x_{ij} = 0, 1, 2, \dots$ не оказывает влияния на значение целевой функции (4.9).

Если затраты, связанные с производством одного изделия, не одинаковы для различных предприятий, то они включаются в величину c_{ij} . Если по каким-то причинам некоторое предприятие недоступно для определенного потребителя, то соответствующая величина x_{ij} исключается из задачи или, когда это удобно, c_{ij} принимается равным ∞ . Обычно $c_{ij} \geq 0$, поэтому неравенства (4.11) можно переписать в виде равенств.

Чтобы задача имела допустимое решение, естественно, требуется, чтобы общая мощность поставщиков была, по крайней мере, не меньше общего спроса потребителей, то есть: $\sum_{i=1}^m S_i \geq \sum_{j=1}^n D_j$. Более того, для решения транспортной задачи удобно принять следующее условие:

$$\sum_{i=1}^m S_i = \sum_{j=1}^n D_j. \quad (4.13)$$

Если условие (4.13) не выполняется, его легко обеспечить, если ввести фиктивного потребителя со спросом, равным $\sum_{i=1}^m S_i - \sum_{j=1}^n D_j$. Присвоим этому потребителю номер $n+1$ и примем $c_{i,n+1}=0$.

Так как (4.13) всегда можно обеспечить, условия (4.10) также можно записать в виде равенств. Отсюда модель транспортной задачи принимает вид:

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \Rightarrow \min, \quad (4.14)$$

$$\sum_{j=1}^n x_{ij} = S_i \quad (i = \overline{1, m}), \quad (\text{предложение}) \quad (4.15)$$

$$\sum_{i=1}^m x_{ij} = D_j \quad (j = \overline{1, n}), \quad (\text{спрос}) \quad (4.16)$$

$$x_{ij} = 0, 1, 2, \dots \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (4.17)$$

где S_i и D_j – положительные целые числа, удовлетворяющие условию (4.13).

Нетрудно заметить, что задача о назначениях является частным случаем классической транспортной задачи при $S_i=1$ ($i = \overline{1, n}$), $D_j=1$ ($j = \overline{1, n}$).

Условия транспортной задачи можно записать в виде **матрицы условий** (рис.34), которая будет использована нами в дальнейшем при применении алгоритма оптимизации. В этой матрице каждая строка соответствует одному пункту отправления, а каждый столбец – пункту назначения.

ПО \ ПН	1	2	...	n	Поставки
1	c_{11} x_{11}	c_{12} x_{12}	...	c_{1n} x_{1n}	S_1
2	c_{21} x_{21}	c_{22} x_{22}	...	c_{2n} x_{2n}	S_2
⋮	⋮	⋮	⋮	⋮	⋮
m	c_{m1} x_{m1}	c_{m2} x_{m1}	...	c_{mn} x_{mn}	S_m
Спрос	D_1	D_2	...	D_n	$\sum_i S_i = \sum_j D_j$

Рис. 34. Матрица условий классической транспортной задачи

4.4.2. Транспортная задача с промежуточными пунктами

Большое практическое значение имеет обобщение классической транспортной модели на случай, когда некоторые пункты служат для перевалки грузов. Например, многие предприятия снабжают товарами магазины не непосредственно, а через оптовые базы, которые могут рассматриваться как промежуточные пункты.

Рассмотрим несколько иную задачу с промежуточными пунктами. На рис.35 представлена схематическая карта в виде сети. Узлы сети соответствуют восьми складам. Положительное число, стоящее у склада, обозначает избыток запасов на складе, который должен быть распределен в системе. Отрицательное число обозначает потребность склада в дополнительных запасах. Таким образом, на складах 1 и 4 избыток запасов, а на складах 3, 6 и 8 требуются дополнительные запасы. Запасы на складах 2, 5 и 7 должны оставаться неизменными. Заметим, что распределенный ресурс может транспортироваться через склады 2, 4, 5, 6 и 7, поэтому эти склады называются *промежуточными*. Склад 1 является источником, а склады 3 и 8 стоками. Необходимо так перераспределить запасы на складах, чтобы общие транспортные расходы при этом были минимальными.

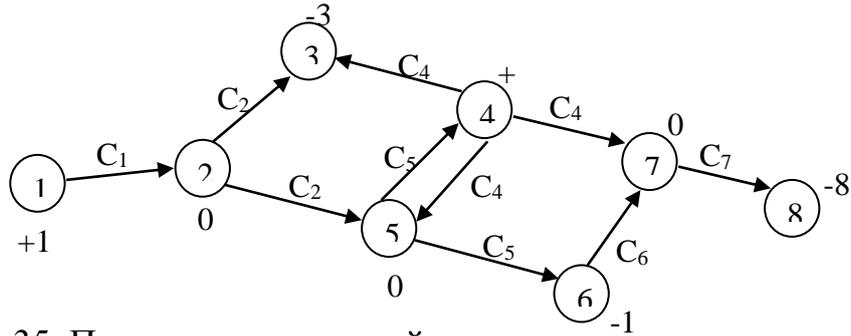


Рис. 35. Пример транспортной сети с промежуточными пунктами

Для нахождения оптимального решения транспортной задачи с промежуточными пунктами можно использовать алгоритм нахождения решения классической транспортной задачи. Для этого предварительно строится матрица классической транспортной задачи следующим образом: каждая строка матрицы соответствует пункту с избытком запасов, а столбец – пункту с недостатком запасов. Для приведенного примера соответствующая матрица приведена на рис. 36. В этой матрице c'_{ij} есть минимальные затраты на перемещение единицы груза из пункта i в пункт j . Таким образом, прежде чем строить данную матрицу, необходимо найти кратчайший путь из каждого узла, в котором имеется избыток запасов, в каждый узел, где их недостаток.

ПО \ ПН	3	6	8	Поставки
1	$\frac{c'_{13}}{X_{13}}$	$\frac{c'_{16}}{X_{16}}$	$\frac{c'_{18}}{X_{18}}$	10
4	$\frac{c'_{43}}{X_{43}}$	$\frac{c'_{46}}{X_{46}}$	$\frac{c'_{48}}{X_{48}}$	2
Спрос	3	1	8	12

Рис. 36. Матрица условий, соответствующая сети на рис. 35

После того, как классическая транспортная задача решена, нетрудно получить решение исходной задачи с промежуточными пунктами. При этом значение каждой переменной задачи с промежуточными пунктами определяется суммой перевозок по тем маршрутам классической транспортной задачи, в которые входит соответствующая дуга исходной задачи.

4.4.3. Метод потенциалов

Данный метод может рассматриваться как упрощенный симплексный метод, предназначенный для нахождения оптимального решения классической транспортной задачи:

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \Rightarrow \min, \quad (4.18)$$

$$\sum_{j=1}^n x_{ij} = S_i \quad (i = \overline{1, m}), \quad (4.19)$$

$$\sum_{i=1}^m x_{ij} = D_j \quad (j = \overline{1, n}), \quad (4.20)$$

$$x_{ij} = 0, 1, 2, \dots \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (4.21)$$

$$\text{где } \sum_{i=1}^m S_i = \sum_{j=1}^n D_j. \quad (4.22)$$

Так как условие (4.22) имеет место, то одно из ограничений (4.19), (4.20) является избыточным. Оно заведомо выполняется, если выполняются остальные $(m+n-1)$ ограничений. Поэтому в модель входят только $(m+n-1)$ независимых ограничений, такое же число переменных входит в любое базисное решение.

Запишем задачу, двойственную по отношению к задаче (4.18) – (4.22):

$$\sum_{i=1}^m S_i v_i + \sum_{j=1}^n D_j \omega_j \Rightarrow \max, \quad (4.23)$$

$$v_i + \omega_j \leq c_{ij} \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (4.24)$$

где величины v_i и ω_j не ограничены по знаку.

Из теоремы о дополнительной нежесткости (см. п.3.6.4) следует, что если выполняются 3 условия:

- 1) величины x_{ij}^* при всех (i, j) удовлетворяют (4.19) – (4.21);
- 2) v_i^* и ω_j^* удовлетворяют (4.24);
- 3) $x_{ij}^*(v_i^* + \omega_j^* - c_{ij}) = 0$ при всех (i, j) ,

то набор величин x_{ij}^* является оптимальным решением транспортной задачи (4.18)-(4.22). В алгоритме, рассматриваемом ниже, на каждой итерации выполняются условия 1 и 3. Выполнение алгоритма заканчивается, как только будет выполнено условие 2.

Метод потенциалов включает те же шаги, что и симплексный алгоритм. Сущность этого метода рассмотрим на примере классической транспортной задачи, матрица условий которой приведена на рис. 37.

ПО \ ПН	1	2	3	4	Поставки
1	<u>19</u> 5	<u>30</u> 2	<u>50</u>	<u>10</u>	7
2	<u>70</u>	<u>30</u> 6-4	<u>40</u> 3+4	<u>60</u>	9
3	<u>40</u>	<u>8</u> +4	<u>70</u> 4-4	<u>20</u> 14	18
Спрос	5	8	7	14	34

Рис. 37. Исходная матрица условий для метода потенциалов

Шаг 1. Необходимо выявить любой допустимый базис исходной задачи (4.18) – (4.22), состоящий из $(m+n-1)$ переменных. Для выполнения данного шага будем использовать так называемый *метод северо-западного угла*. Название метода обусловлено тем, что распределение поставок по потребителям начинается с верхнего левого угла, который на географических картах соответствует северо-западу. При этом сначала полностью распределим мощность первого поставщика, затем – второго и так далее. Элементы полученного таким образом исходного базиса поместим в соответствующие клетки матрицы условий (рис.37). (Не обращайтесь пока внимание на выделенные числа и предшествующие им знаки «+» и «-»).

Начальное значение целевой функции:

$$19 \cdot 5 + 30 \cdot 2 + 30 \cdot 6 + 40 \cdot 3 + 70 \cdot 4 + 20 \cdot 14 = 1015.$$

Итерация 1

Шаг 2. Ищется такая небазисная переменная x_{ij} , за счет включения которой в базис можно улучшить значение целевой функции.

Идея выполнения данного шага заключается в следующем. Во-первых, находится пробное решение двойственной задачи путем решения системы $m+n-1$ ограничений двойственной задачи, каждое из которых соответствует текущей базисной переменной исходной задачи. То есть решается система уравнений $v_i + w_j = c_{ij}$ (для каждой базисной переменной) и таким образом обеспечивается выполнение условия дополнительной нежесткости.

Во-вторых, для каждой небазисной переменной x_{ij} ищется значение: $v_i + w_j - c_{ij}$. Если каждое подобное значение не положительно, то, следовательно, пробное решение двойственной задачи является допустимым и удовлетворяет ограничениям (4.24). В этом случае исходное базисное решение является оптимальным. Если указанная величина для какой-то небазисной переменной положительна, то соответствующая переменная является кандидатом на

включение в следующий базис. Если таких переменных несколько, из них в базис включается та, которой соответствует наибольшее значение $v_i + \omega_j - c_{ij}$.

Рассмотрим реализацию данного шага для нашего примера. Во-первых, запишем систему $m+n-1$ ограничений двойственной задачи, каждое из которых соответствует своей базисной переменной x_{ij} :

$$\begin{array}{rcll} v_1 & + \omega_1 & = 19, & \text{маршрут (1,1),} \\ v_1 & & + \omega_2 & = 30, \quad \text{--/-- (1,2),} \\ v_2 & & + \omega_2 & = 30, \quad \text{--/-- (2,2),} \\ v_2 & & & + \omega_3 & = 40, \quad \text{--/-- (2,3),} \\ v_3 & & & + \omega_3 & = 70, \quad \text{--/-- (3,3),} \\ v_3 & & & & + \omega_4 & = 20, \quad \text{--/-- (3,4).} \end{array}$$

Заметим, что это система из шести уравнений с семью неизвестными. Для получения решения необходимо выбрать одну переменную, и приписать ей произвольное значение. Допускается выбор любой переменной, и разрешается приписывать ей любое удобное значение. Возьмем, например, v_1 , так как эта переменная входит в два соотношения, и примем $v_1=0$. Отсюда: $\omega_1=19$, $\omega_2=30$, $v_2=0$, $\omega_3=40$, $v_3=30$, $\omega_4=-10$.

Теперь получим оценку каждого маршрута, не входящего в базис, по формуле $v_i + \omega_j - c_{ij}$:

$$\begin{array}{rcll} 0 + 40 - 50 & = & -10, & \text{маршрут (1,3),} \\ 0 - 10 - 10 & = & -20, & \text{--/-- (1,4),} \\ 0 + 19 - 70 & = & -51, & \text{--/-- (2,1),} \\ 0 - 10 - 60 & = & -70, & \text{--/-- (2,4),} \\ 30 + 19 - 40 & = & 9, & \text{--/-- (3,1),} \\ 30 + 30 - 8 & = & 52, & \text{--/-- (3,2).} \end{array}$$

Отсюда видно, что в базис следует включить переменную x_{32} .

Данный шаг выполняется быстрее, если использовать **матрицу оценок** (рис. 38), похожую на матрицу условий. В этой матрице сначала заполняются клетки, соответствующие базисным маршрутам. Во все эти клетки ставится цифра 0. Затем записывается любое значение какой-то одной двойственной переменной. Затем проставляются значения всех остальных двойственных переменных такие, чтобы для всех клеток с 0 выполнялось: $v_i + \omega_j = c_{ij}$. После этого во всех клетках, соответствующих небазисным переменным записывается $v_i + \omega_j - c_{ij}$.

ПО \ ПН	1	2	3	4	v_i
1	$\frac{19}{0}$	$\frac{30}{0}$	$\frac{50}{-10}$	$\frac{10}{-20}$	0
2	$\frac{70}{-51}$	$\frac{30}{0}$	$\frac{40}{0}$	$\frac{60}{-70}$	0
3	$\frac{40}{9}$	$\frac{8}{52}$	$\frac{70}{0}$	$\frac{20}{0}$	30
ω_j	19	30	40	-10	

Рис. 38. Матрица оценок для итерации 1

Шаг 3. Находится тот маршрут, который исключается из базиса.

Поставим знак «+» в той клетке матрицы условий, которая соответствует вновь вводимой переменной, в примере это клетка (3,2) (рис.37). Так как мы будем брать для нового маршрута продукцию у поставщика 3, мы должны уменьшить на эту же величину какую-то прежнюю поставку от того же поставщика. Это вызовет необходимость скомпенсировать спрос соответствующего потребителя за счет другого поставщика и т.д. В результате мы должны выявить на матрице условий замкнутый контур (цикл), который содержит вновь вводимую переменную x_{32} . Каждый элемент цикла со знаком «+» обязательно компенсируется двумя элементами со знаком «-», один из которых записывается в этой же строке, а второй – в этом же столбце. Цикл (контур) перераспределения не обязательно должен быть четырехугольным, но все его углы обязательно должны быть прямыми.

После того, как в матрице условий обозначен цикл из знаков «+» и «-», необходимо определить предельную величину перераспределения по данному циклу. Эта величина определяется наименьшим содержимым клетки со знаком «-». В рассматриваемом примере (см. рис.37) первой обращается в 0 переменная x_{33} . Ее мы и исключим из базиса.

Шаг 4. Определяются значения переменных нового базиса.

Содержимое клеток матрицы условий со знаком «+» увеличивается, а со знаком «-» уменьшается на величину переменной, исключенной из базиса. В примере эта величина равна 4 (см. рис. 37). Значения переменных нового базиса: $x_{11}=5$, $x_{12}=2$, $x_{22}=2$, $x_{23}=7$, $x_{32}=4$, $x_{34}=14$. Целевая функция равна: $19 \cdot 5 + 30 \cdot 2 + 30 \cdot 2 + 40 \cdot 7 + 8 \cdot 4 + 20 \cdot 14 = 807$.

Значения переменных нового базиса поместим в матрицу условий для следующей итерации (рис.39). (Не обращайте внимание на выделенные числа.)

ПО \ ПН	1	2	3	4	Поставки
1	$\frac{19}{5}$	$\frac{30}{2-2}$	$\frac{50}{7}$	$\frac{10}{+2}$	7
2	$\frac{70}{-51}$	$\frac{30}{0}$	$\frac{40}{0}$	$\frac{60}{-18}$	9
3	$\frac{40}{-43}$	$\frac{8}{0}$	$\frac{70}{-52}$	$\frac{20}{0}$	18
Спрос	5	8	7	14	34

Рис. 39. Матрица условий для итерации 2

Итерация 2

Шаг 2. Ищется такая небазисная переменная x_{ij} , за счет включения которой в базис можно улучшить значение целевой функции. Для этого на основе матрицы условий (см. рис.39) запишем соответствующую матрицу оценок (рис.40). Из нее видно, что в базис следует включить x_{14} .

ПО \ ПН	1	2	3	4	v_i
1	$\frac{19}{0}$	$\frac{30}{0}$	$\frac{50}{-10}$	$\frac{10}{32}$	0
2	$\frac{70}{-51}$	$\frac{30}{0}$	$\frac{40}{0}$	$\frac{60}{-18}$	0
3	$\frac{40}{-43}$	$\frac{8}{0}$	$\frac{70}{-52}$	$\frac{20}{0}$	-22
ω_j	19	30	40	42	

Рис. 40. Матрица оценок для итерации 2

Шаг 3. Находится тот маршрут, который исключается из базиса.

Запишем в клетку (1,4) матрицы условий знак «+» и выявим соответствующий цикл перераспределения (см. рис.39). Из рассмотрения этого цикла следует, что из базиса следует исключить x_{12} .

Шаг 4. Определяются значения переменных нового базиса.

Значения переменных нового базиса: $x_{11}=5$, $x_{14}=2$, $x_{22}=2$, $x_{23}=7$, $x_{32}=6$, $x_{34}=12$. Целевая функция равна:

$$19 \cdot 5 + 10 \cdot 2 + 30 \cdot 2 + 40 \cdot 7 + 8 \cdot 6 + 20 \cdot 12 = 743.$$

Значения переменных нового базиса поместим в матрицу условий для следующей итерации (рис. 41).

ПО \ ПН	1	2	3	4	Поставки
1	$\frac{19}{5}$	$\frac{30}{}$	$\frac{50}{}$	$\frac{10}{2}$	7
2	$\frac{70}{}$	$\frac{30}{2}$	$\frac{40}{7}$	$\frac{60}{}$	9
3	$\frac{40}{}$	$\frac{8}{6}$	$\frac{70}{}$	$\frac{20}{12}$	18
Спрос	5	8	7	14	34

Рис.41. Матрица условий для итерации 3

Итерация 3

Шаг 2. Ищется такая небазисная переменная x_{ij} , за счет включения которой в базис можно улучшить значение целевой функции. Для этого на основе матрицы условий (см. рис.41) запишем соответствующую матрицу оценок (рис. 42).

ПО \ ПН	1	2	3	4	v_i
1	$\frac{19}{0}$	$\frac{30}{-32}$	$\frac{50}{-42}$	$\frac{10}{0}$	0
2	$\frac{70}{-19}$	$\frac{30}{0}$	$\frac{40}{0}$	$\frac{60}{-18}$	32
3	$\frac{40}{-11}$	$\frac{8}{0}$	$\frac{70}{-52}$	$\frac{20}{0}$	10
ω_j	19	-2	8	10	

Рис. 42. Матрица оценок для итерации 3

Из матрицы оценок видно, что оценки $v_i + \omega_j - c_{ij}$ всех небазисных маршрутов отрицательны и, следовательно, оптимальное решение получено:

$x_{11}^* = 5$, $x_{14}^* = 2$, $x_{22}^* = 2$, $x_{23}^* = 7$, $x_{32}^* = 6$, $x_{34}^* = 12$, оптимальное значение целевой функции 743.

4.4.4. Устранение вырожденности

В приведенном выше описании метода потенциалов предполагалось, что полученное на каждой итерации базисное решение является **невыврожденным**, т.е. состоящим из $m+n-1$ ненулевых переменных. На практике это часто

не выполняется. Если выявленный на текущей итерации базис является вырожденным, то возникают трудности с нахождением значений двойственных переменных, так как количество уравнений, соответствующих базисным переменным вида $u_i + a_j - c_{ij} = 0$, уменьшается на одно. Для устранения вырожденности можно прибегнуть к следующему приему.

Значение одной из переменных, равное нулю, считается теперь равное ε ($\varepsilon > 0$). На величину ε изменяются также соответствующая потребность и соответствующий спрос. ε - очень малая величина, обладающая следующими свойствами: 1) $\varepsilon \ll 1$; 2) $\varepsilon + 0 = \varepsilon$; 3) $x_{ij} \pm \varepsilon \approx x_{ij}$, $x_{ij} > 0$. Так как ε - искусственная величина, введенная для удобства, на последней итерации принимается $\varepsilon = 0$.

Пример. Пусть транспортная задача описывается матрицей условий, приведенной на рис. 43 (без учета выделенных символов). Исходный базис, полученный на шаге 1 методом северо-западного угла, является вырожденным, так как в него входят не 5, а 4 ненулевые переменные. Вырожденность следует из того, что ненулевая базисная переменная x_{33} является единственной как в своей строке, так и в своем столбце.

ПО \ ПН	1	2	3	Поставки
1	$\begin{array}{ c } \hline 7 \\ \hline \end{array}$ $2 - \varepsilon$	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$ $+\varepsilon$	2
2	$\begin{array}{ c } \hline 2 \\ \hline \end{array}$ $2 + \varepsilon$	$\begin{array}{ c } \hline 1 \\ \hline \end{array}$ 1	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$ $\varepsilon - \varepsilon$	$3 + \varepsilon$
3	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$	$\begin{array}{ c } \hline 6 \\ \hline \end{array}$ 5	5
Спрос	4	1	$5 + \varepsilon$	10

Рис. 43. Матрица условий с вырожденным базисом

Для устранения вырожденности значение любой нулевой переменной в строке 3 или в столбце 3 заменим на ε . Сделаем это для x_{23} (рис. 43). Кроме того, увеличим на ε мощность поставщика 2 и спрос потребителя 3.

Итерация 1

Шаг 2. Ищется такая небазисная переменная x_{ij} , за счет включения которой в базис можно улучшить значение целевой функции. Для определения этой переменной запишем матрицу оценок (рис. 44). Из этой матрицы следует, что в базис следует включить x_{13} .

ПО \ ПН	1	2	3	u_i
1	$\begin{array}{ c } \hline 7 \\ \hline \end{array}$ 0	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$ 3	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$ 4	5
2	$\begin{array}{ c } \hline 2 \\ \hline \end{array}$ 0	$\begin{array}{ c } \hline 1 \\ \hline \end{array}$ 0	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$ 0	0
3	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$ 2	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$ 0	$\begin{array}{ c } \hline 6 \\ \hline \end{array}$ 0	3
ω_j	2	1	3	

Рис. 44. Матрица оценок для итерации 1

Шаг 3. Находится тот маршрут, который исключается из базиса.

Запишем в клетку (1,3) матрицы условий знак «+» и выявим соответствующий цикл перераспределения (см. рис. 43). Из рассмотрения этого цикла следует, что из базиса следует исключить x_{23} .

Шаг 4. С учетом свойств ε определяются значения переменных нового базиса, которые помещаются в матрицу условий для следующей итерации 2 (рис. 45). (Не обращайтесь пока внимание на выделенные символы.)

ПО \ ПН	1	2	3	Поставки
1	$\begin{array}{ c } \hline 7 \\ \hline \end{array}$ 2-2	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$ $\varepsilon+2$	2
2	$\begin{array}{ c } \hline 2 \\ \hline \end{array}$ 2	$\begin{array}{ c } \hline 1 \\ \hline \end{array}$ 1	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$	3
3	$\begin{array}{ c } \hline 3 \\ \hline \end{array}$ +2	$\begin{array}{ c } \hline 4 \\ \hline \end{array}$	$\begin{array}{ c } \hline 6 \\ \hline \end{array}$ 5-2	5
Спрос	4	1	5	10

Рис. 45. Матрица условий для итерации 2

Итерация 2

Шаг 2. Для определения новой базисной переменной запишем матрицу оценок (рис. 46). Из этой матрицы следует, что в базис следует включить x_{31} .

ПО \ ПН	1	2	3	u_i
1	7 0	3 3	4 0	5
2	2 0	1 0	3 -4	0
3	3 6	4 4	6 0	7
ω_j	2	1	-1	

Рис. 46. Матрица оценок для итерации 2

Шаг 3. Из базиса следует исключить x_{11} .

Шаг 4. Запишем матрицу условий для итерации 3 (рис.47). Так как число ненулевых переменных в этой матрице 5 и, следовательно, базис невырожденный, далее метод потенциалов применяется обычным образом.

ПО \ ПН	1	2	3	Поставки
1	7 2	3 1	4 2	2
2	2 2	1 1	3 3	3
3	3 2	4 4	6 3	5
Спрос	4	1	5	10

Рис. 47. Матрица условий для итерации 3

4.4.5. Анализ классической транспортной задачи на чувствительность

Как и при проведении анализа на чувствительность общей задачи линейного программирования, при анализе транспортной модели очень полезно понятие двойственности. Рассмотрим следующие три частные задачи анализа на чувствительность.

Изменение коэффициента в целевой функции при небазисной переменной. Естественно, что при увеличении данного коэффициента прежнее решение останется оптимальным. При уменьшении же коэффициента ниже некоторого предельного значения прежний базис должен быть скорректиро-

ван за счет включения в него данной переменной. Данное предельное значение можно найти из следующих соображений. Так как на последней итерации оценки всех небазисных переменных удовлетворяют условию: $v_i + \omega_j - c_{ij} \leq 0$, то соответствующая величина в матрице оценок показывает, насколько может быть изменен соответствующий коэффициент c_{ij} без нарушения оптимальности прежнего решения.

Например, для рассмотренного ранее примера применения метода потенциалов на заключительной итерации получили матрицу условий (см. рис. 41) и матрицу оценок (см. рис.42), из которой видно, что коэффициент c_{13} может быть изменен на 42 единицы с 50 до 8. Аналогично c_{33} может быть уменьшен с 70 до 8. При большем уменьшении c_{33} прежнее решение будет неоптимальным.

Увеличение мощности любого поставщика на единицу. Допустим, что увеличилась мощность S_p какого-то поставщика p . Естественно предположить, что значение целевой функции при этом не ухудшится, то есть не увеличится. Так как оптимальное значение целевой функции исходной задачи равно оптимальному значению целевой функции двойственной задачи, то для определения изменения первой из этих функций можно рассмотреть изменение второй. Целевая функция двойственной задачи имеет вид:

$$\sum_{i=1}^m S_i v_i + \sum_{j=1}^n D_j \omega_j \Rightarrow \max. \quad (4.25)$$

Так как S_p увеличилась на единицу, целевая функция (4.25) увеличилась на v_p^* , где v_p^* - оптимальное значение переменной v_p двойственной задачи. Так как суммарная мощность поставщиков равна сумме спросов, то увеличение поставок от источника p должно сопровождаться соответствующим уменьшением поставок от других поставщиков. Всего выгоднее уменьшить поставки от того поставщика, которому соответствует наибольшее значение v^* .

Таким образом, можно сделать следующий вывод: при увеличении S_p на единицу, значение целевой функции уменьшится на величину $(\max_i (v_i^*) - v_p^*) \geq 0$.

Например, для рассмотренного ранее примера применения метода потенциалов при увеличении S_1 с 7 до 8, целевая функция уменьшится на $(32-0)=32$. Покажем, что это так, воспользовавшись матрицей условий на заключительной итерации (рис.48). Во-первых, естественно выделить новую единицу мощности на увеличение x_{14} (так как $c_{14} < c_{11}$). Теперь $x_{14}=3$. Поток x_{34} необходимо уменьшить на единицу, так как D_4 по-прежнему равно 14. Освободившуюся единицу S_3 можно направить на увеличение x_{32} , что позволит

уменьшить на единицу x_{22} . Общее приращение целевой функции будет: $10-20+8-30 = -32$. Заметим, что неиспользованная единица мощности обязательно приходится на источник, которому соответствует $\max(v_i^*)$ (в примере - пункт отправления 2). Кроме того, обратим внимание, что контур перераспределения в данном случае не является замкнутым. Этот контур начинается в строке матрицы условий с увеличенной поставкой и заканчивается в строке с уменьшенной поставкой.

ПО \ ПН	1	2	3	4	Поставки
1	<u>19</u> 5	<u>30</u>	<u>50</u>	<u>10</u> 2+1	7+1
2	<u>70</u>	<u>30</u> 2-1	<u>40</u> 7	<u>60</u>	9 (-1)
3	<u>40</u>	<u>8</u> 6+1	<u>70</u>	<u>20</u> 12-1	18
Спрос	5	8	7	14	34

Рис. 48. Матрица условий для последней итерации при увеличении S_1

Воспользовавшись матрицей условий на рис. 48, нетрудно найти предельное увеличение δ_1 ($\delta_1 > 0$) мощности S_1 поставщика 1, при которой прежний план остается оптимальным. Для этого на рис.48 достаточно заменить приращение «1» на приращение « δ_1 ». Предельное значение δ_1 определяется клеткой (2,2) и составляет 2 единицы.

Одновременное увеличение на единицу мощности любого поставщика S_i и спроса любого потребителя D_j . Из выражения для целевой функции двойственной задачи (4.25) видно, что ее оптимальное значение увеличится на величину $v_i + w_j$. На эту же величину естественно увеличится и значение целевой функции исходной задачи.

Например, для рассматриваемого примера при $S_1=8$ и $D_2=9$ целевая функция увеличится на $(0-2) = -2$ единицы, то есть несмотря на увеличение суммарного объема перевозок, целевая функция даже улучшилась. Рассмотрим прошедшие при этом изменения в решении исходной задачи (рис. 49). Избыточную единицу S_1 направим на увеличение x_{14} до 3-х. Это приведет к уменьшению на 1 x_{34} , и, следовательно, к увеличению на 1 x_{32} . Изменения при этом заканчиваются, так как увеличение x_{32} до 7 соответствует увеличенному до 9 значению D_2 . Приращение целевой функции составит: $10-20+8 = -2$.

ПО \ ПН	1	2	3	4	Поставки
1	19 5	30	50	10 2+1	7+1
2	70	30 2	40 7	60	9
3	40	8 6+1	70	20 12-1	18
Спрос	5	8+1	7	14	34

Рис. 49. Матрица условий для последней итерации при увеличении S_1 и D_2

Воспользовавшись матрицей условий на рис. 49, нетрудно найти предельное увеличение δ ($\delta > 0$) мощности S_1 и спроса D_2 , при которой прежний план остается оптимальным. Для этого на рис. 49 достаточно заменить приращение «1» на приращение « δ ». Предельное значение δ определяется клеткой (3,4) и составляет 12 единиц.

4.5. Задача коммивояжера

4.5.1. Описание модели

Краткое описание задачи коммивояжера следующее. Коммивояжер (агент по сбыту) собирается посетить каждый из n городов по одному разу, выехав из города 1 и вернувшись в него же. Расстояние между городом i и городом j равно c_{ij} . Требуется найти кратчайший маршрут коммивояжера. Условия данной задачи могут быть заданы с помощью соответствующей сети. Пример такой сети приведен на рис. 50.

Следует отметить, что подобным образом могут рассматриваться некоторые задачи, не имеющие ничего общего с перемещением на местности. Например, такой же сетевой моделью можно описать условия производства мороженого различных сортов на одном и том же оборудовании. Пусть c_{ij} есть затраты времени на очистку и подготовку оборудования, если сорт j изготавливается после сорта i . Величины c_{ij} могут изменяться в широком диапазоне, так как сорта, мало отличающиеся друг от друга (например, шоколадное и кофейное) можно выпускать друг за другом, лишь незначительно изменив технологию, тогда как для последовательного выпуска некоторых других сортов требуется много времени на очистку оборудования (например, если за шоколадным следует фруктовое). Требуется определить такой порядок выпуска различных сортов в течение дня, чтобы минимизировать общие затраты времени на переналадку.

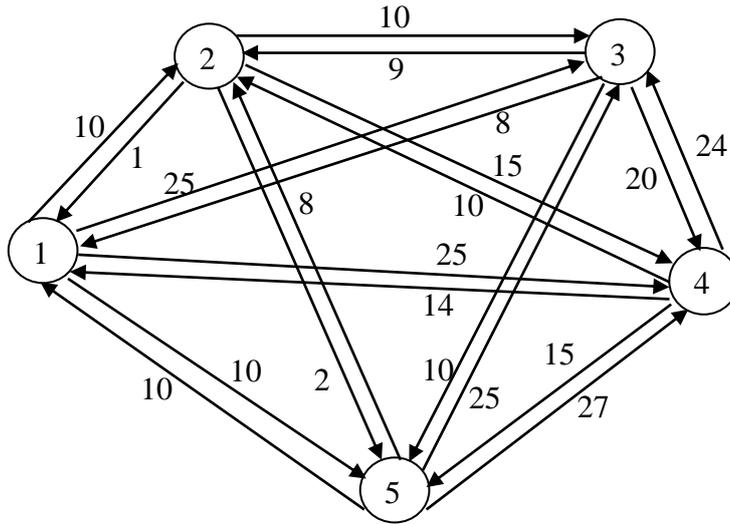


Рис. 50. Пример сети задачи коммивояжера

Для записи соответствующей математической модели введем управляемые переменные x_{ij} :

$$x_{ij} = \begin{cases} 1, & \text{если в маршрут входит переезд из города } i \text{ в город } j \text{ } (i \neq j), \\ 0, & \text{в остальных случаях,} \end{cases}$$

где $i = \overline{1, n}$, $j = \overline{1, n}$.

Математическая модель имеет вид:

$$\sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \Rightarrow \min, \quad (4.26)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad (i = \overline{1, n}), \quad (4.27)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad (j = \overline{1, n}), \quad (4.28)$$

$$x_{ij} = 0, 1 \quad (i = \overline{1, n}, j = \overline{1, n}), \quad (4.29)$$

$$\text{решение есть цикл.} \quad (4.30)$$

Без условия (4.30), которое записано вообще говоря не в математической форме, совокупности условий (4.27) и (4.28), которые описывают соответственно, обязательный въезд коммивояжера в каждый город, причем один раз, и аналогично выезд из него, могут удовлетворять маршруты, каждый из которых есть набор несвязанных подциклов. Например, один из подциклов включает только пункты 1 и 2, а второй – все остальные.

4.5.2. Алгоритм решения задачи коммивояжера

Данный алгоритм является модификацией широко распространенного *метода ветвей и границ*, сущность которого состоит в следующем. Исходная сложная в решении задача заменяется совокупностью более простых независимых друг от друга задач, каждая из которых может быть заменена совокупностью других задач и так далее. Таким образом, в процессе применения данного метода строится дерево задач, корневая вершина которого соответствует исходной задаче. В процессе построения дерева, прежде чем заменять какую-то задачу совокупностью более простых задач, для данной задачи ищется такое граничное значение целевой функции, улучшить которое никакое решение данной задачи заведомо не сможет. Если это граничное значение хуже, чем какое-то уже найденное допустимое решение на других ветвях дерева задач, то данная задача больше не рассматривается. Процесс построения дерева заканчивается, как только будет найдено решение, улучшить которое ни одна из задач, соответствующих висячим вершинам дерева заведомо уже не может.

Обычно метод поиска лучшей границы оптимального значения целевой функции рассматриваемой задачи заключается в том, что в модели, соответствующей данной задаче, отбрасывается какое-то ограничение. Так как в качестве такого ограничения обычно берется самое неудобное, то процесс решения измененной модели существенно упрощается. Что касается оптимального значения целевой функции, то исключение из модели каких-то ограничений может это значение только улучшить, но не наоборот.

Алгоритм решения задачи (4.26) – (4.30) включает следующие шаги.

Шаг 1. Исходная задача заносится в дерево задач. Берется любое допустимое решение задачи, например, $x_{12}=x_{23}=x_{34} \dots =x_{n-1,n}=x_{n1}=1$. Соответствующее значение целевой функции берется в качестве x_{OB}^1 - верхней (худшей) границы оптимального значения целевой функции на 1-й итерации.

Шаг 2. Прекратить вычисления, если нерассмотренных ранее висячих вершин дерева задач больше нет. Иначе – выбрать одну из данных задач.

Шаг 3. Сущность данного шага состоит в том, что на нем находится нижняя (лучшая) граница x_{OH} для оптимального значения целевой функции задачи, выбранной ранее на шаге 2. Для поиска x_{OH} рассматриваемая задача заменяется соответствующей задачей о назначениях. Такая замена выполняется очень просто: отбрасыванием ограничения, требующего того, чтобы решение было циклом. В самом деле, задача (4.26) – (4.30) включает в себя формальную запись задачи о назначениях (4.26) – (4.29) плюс требование (4.30) о том, что решение должно быть циклом. Отбрасывание ограничения (4.30) может только улучшить значение целевой функции.

Если $x_{OH} \geq x_{OB}^t$, то принять $x_{OB}^{t+1} = x_{OB}^t$ и вернуться к шагу 2. Иначе - переход на шаг 4.

Шаг 4. Если полученное оптимальное решение задачи о назначениях является циклом, то зафиксировать его в качестве верхней границы x_{OB}^{t+1} целевой функции и вернуться на шаг 2. Иначе $x_{OB}^t \rightarrow x_{OB}^{t+1}$ и переход на шаг 5.

Шаг 5. Рассмотреть в полученном оптимальном решении задачи о назначениях подцикл, содержащий минимальное число вершин. Каждой дуге (i,j) в выбранном подцикле поставить в соответствие задачу коммивояжера, отличающуюся от задачи, выбранной на шаге 2 тем, что теперь $c_{ij}=\infty$. Все полученные задачи внести в дерево задач в качестве «дочерних» задачи, выбранной ранее на шаге 2. Переход на шаг 2.

Пример. Детали алгоритма поясним на примере задачи коммивояжера, сеть которой приведена на рис.50.

Шаг 1. Исходная задача заносится в дерево задач в качестве задачи 1 (рис.51). В качестве исходного допустимого решения берем решение: $x_{12}=x_{23}=x_{34}=x_{45}=x_{51}=1$. Соответствующее значение целевой функции возьмем в качестве верхней границы оптимального значения целевой функции на первой итерации:

$$x_{OB}^1 = 10 + 10 + 20 + 15 + 10 = 65.$$

Это значение x_{OB}^1 запишем в прямоугольник, соответствующий задаче 1 (рис. 51).

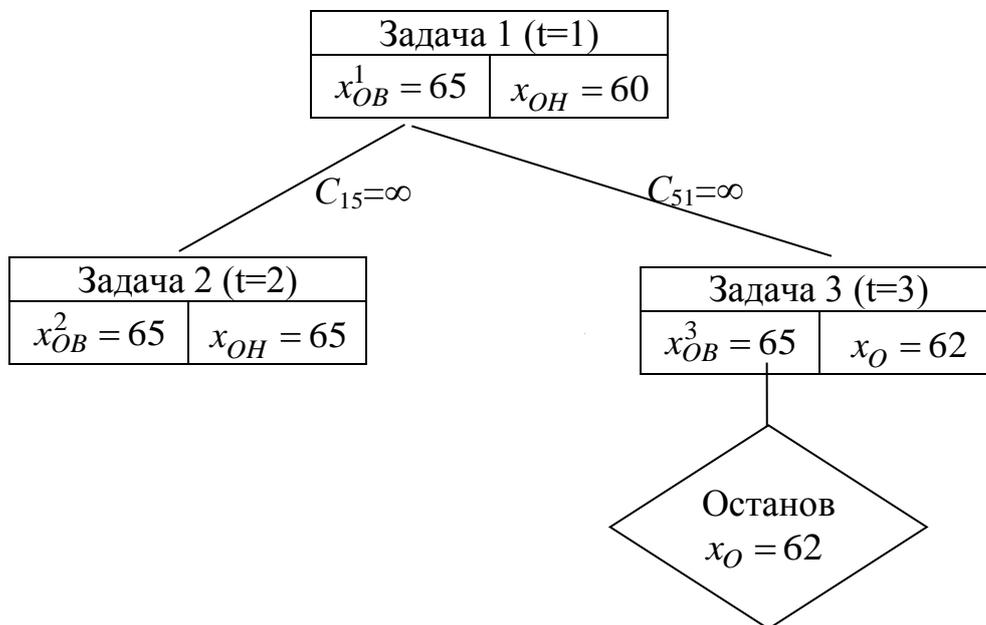


Рис. 51. Дерево задач для примера задачи коммивояжера

Итерация 1

Шаг 2. Выбираем единственную висячую вершину дерева задач – задачу 1.

Шаг 3. Решаем задачу о назначениях, соответствующую задаче 1. Для этого сведем в матрицу условия задачи о назначениях (рис.52).

Полученное решение имеет вид: $x_{15}=x_{51}=x_{23}=x_{34}=x_{42}=1$, $x_{OH} = 60$. На рис. 53 данное решение изображено графически. Запишем значение $x_{OH}=60$ в прямоугольник для задачи 1 (рис. 51).

∞	10	25	25	10
1	∞	10	15	2
8	9	∞	20	10
14	10	24	∞	15
10	8	25	27	∞

Рис. 52. Матрица задачи о назначениях на 1-й итерации

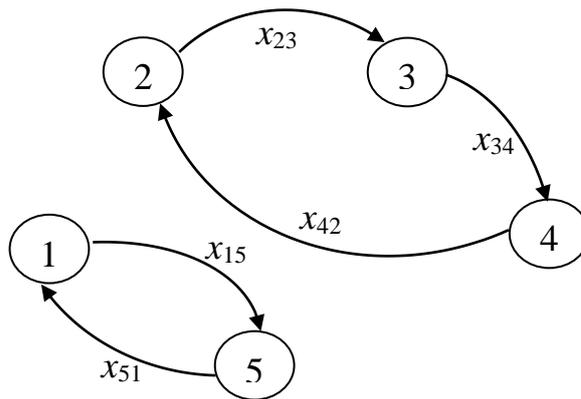


Рис. 53. Графическое изображение решения задачи о назначениях на 1-й итерации

Шаг 4. Полученное оптимальное решение задачи о назначениях не является циклом, поэтому $x_{OB}^1 \rightarrow x_{OB}^2$ и переход на шаг 5.

Шаг 5. Подциклом с минимальным числом вершин является подцикл 1-5 (см. рис. 53). Поэтому запишем в дерево задач две новые задачи: задачу 2 и задачу 3 (рис.51). Первая из них отличается от задачи 1 тем, что в ней $c_{15}=\infty$, вторая – $c_{51}=\infty$.

Итерация 2

Шаг 2. Выбираем задачу 2. Записываем в соответствующий прямоугольник $t=2$, а также $x_{OB}^2=65$.

Шаг 3. Решаем задачу о назначениях, соответствующую задаче 2. Полученное решение: $x_{OH} = 65$.

Так как $x_{OH} \geq x_{OB}^2$, то принимаем $x_{OB}^3 = x_{OB}^2 = 65$. Переход на шаг 2.

Итерация 3

Шаг 2. Выбираем задачу 3.

Шаг 3. Решаем задачу о назначениях, соответствующую задаче 3. Полученное решение: $x_{15} = x_{52} = x_{23} = x_{34} = x_{41} = 1$, $x_O = 62$. Здесь записано x_O , а не x_{OH} , так как решение – цикл.

Шаг 4. Так как решение задачи о назначениях есть цикл, то $x_{OB}^4 = x_O = 62$.

Итерация 4

Шаг 2. Так как нерассмотренных ранее висячих вершин дерева задач больше нет, то выполнение алгоритма закончено. Оптимальное решение: $x_{15}^* = x_{52}^* = x_{23}^* = x_{34}^* = x_{41}^* = 1$, $x_O^* = x_{OB}^4 = 62$.

4.6. Примеры задач

Задача 4.1.

При составлении расписания использования аудиторий в одном из университетов требуется выделить n аудиторий для n учебных групп. Некоторые возможности исключены по одной из следующих причин: численность группы превышает количество мест в аудитории, в нужное время аудитория занята, аудитория находится на слишком большом расстоянии от преподавательской. Предложите математическую модель, которая может оказать помощь в решении этой задачи.

Решение

Данная задача относится к задачам о назначениях. Вводим управляемые переменные x_{ij} ($i, j = \overline{1, n}$):

$$x_{ij} = \begin{cases} 1, & \text{если группа } i \text{ занимается в аудитории } j, \\ 0, & \text{в противном случае.} \end{cases}$$

Пусть c_{ij} – оценки, полученные обработкой экспертной информации. c_{ij} показывает, каковы потери, если группа i будет заниматься в аудитории j . Для исключения некоторых возможностей, указанных в условии задачи, припишем соответствующим c_{ij} значения, очень большие по величине (например, 1000). Целевая функция модели:

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

где переменные x_{ij} должны удовлетворять соотношениям:

$$\sum_{j=1}^n x_{ij} = 1, \quad i = \overline{1, n}, \quad (\text{группа } i \text{ занимается в одной аудитории)},$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j = \overline{1, n}, \quad (\text{в каждой аудитории } j \text{ занимается одна группа)},$$

$$x_{ij} = 0, 1.$$

Задача 4.2

Фирма является владельцем мелкосерийного металлообрабатывающего завода. Дневной портфель заказов включает n деталей, каждая из которых может обрабатываться на n различных станках. Пусть t_{ij} – общая продолжительность обработки детали i на станке j (включая время наладки станка). Требуется построить оптимизационную модель, минимизирующую общую продолжительность выполнения всех заказов. (Данная задача целесообразна тогда, когда нужно минимизировать затраты рабочей силы.)

Решение

1. Введем управляемые переменные:

$$x_{ij} = \begin{cases} 1 - \text{деталь } i \text{ обрабатывается на } j - \text{ом станке,} \\ 0 - \text{не обрабатывается.} \end{cases}$$

2. Математическая модель

$$\sum_{i=1}^n \sum_{j=1}^n t_{ij} x_{ij} \Rightarrow \min,$$

$$\sum_{i=1}^n x_{ij} = 1, \quad j = \overline{1, n}, \quad (\text{на станке } j \text{ обрабатывается одна деталь}),$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = \overline{1, n}, \quad (\text{каждая деталь } i \text{ обрабатывается на одном станке)},$$

$$x_{ij} = 0, 1, \quad i, j = \overline{1, n}.$$

Задача 4.3

В одном из цехов металлообрабатывающего завода работает дорогостоящий сверлильный станок, который по мере износа необходимо периодически заменять. Вице-президент фирмы, ведающий вопросами производства, разрешил установить новую модель станка, но при этом поручил начальнику цеха составить план замены на ближайшие семь лет, после чего станок больше не понадобится.

Пусть p_t – стоимость новой модели в период t , где $p_1=100$, $p_2=105$, $p_3=110$, $p_4=115$, $p_5=120$, $p_6=125$, $p_7=130$, в связи с чем приобретение станка в рассматриваемый период обходится в 100 единиц. Пусть v_k – остаточная стоимость станка, продаваемого по окончании k периодов эксплуатации, где $v_1=50$, $v_2=25$, $v_3=10$, $v_4=5$, $v_5=2$, $v_6=1$, $v_7=0$. Таким образом, если новый станок продать в конце периода 1, то фирма получит доход в 50 единиц, если продать его в конце периода 2, то доход составит 25 единиц, и наконец, если продать станок в конце периода 7, то доход будет равен нулю. Примем, что r_k – затраты на эксплуатацию станка в течение k -го периода эксплуатации, где $r_1=30$, $r_2=40$, $r_3=r_4=50$, $r_5=60$, $r_6=70$, $r_7=100$.

Требуется построить сетевую модель замены станка.

Решение

Сетевая модель задачи приведена на рис. 54. Каждый кружок соответствует началу соответствующего года планирования. Если дуга входит в кружок, то это означает, что к началу указанного в кружке года станок продается. Исходящая дуга соответствует покупке в начале указанного в кружке года. Например, дуга 1,8 соответствует тому случаю, когда станок служит без замены все 7 лет.

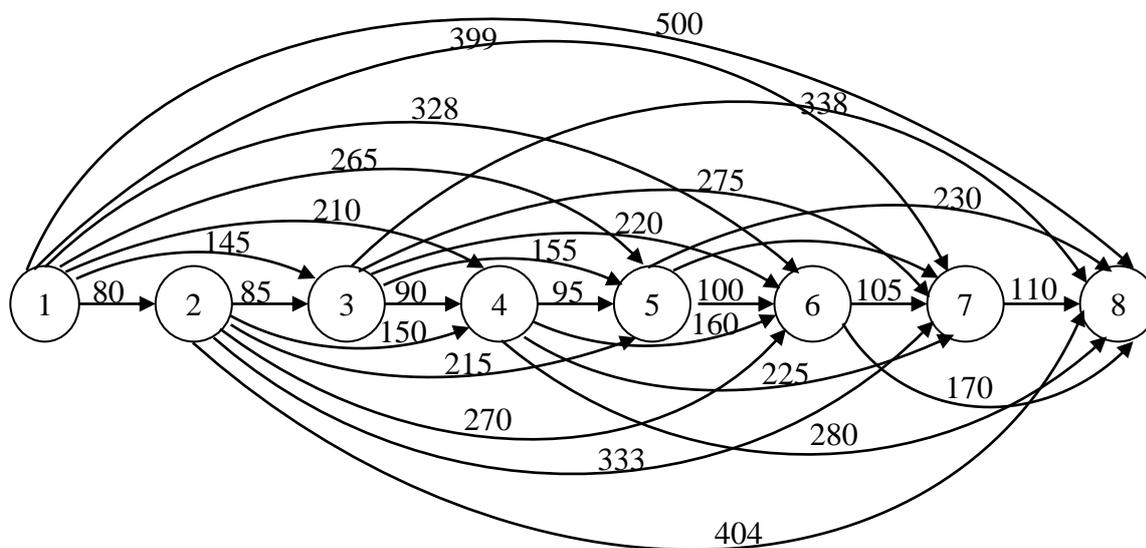


Рис. 54. Ациклическая сеть для задачи 4.3 по замене станка

Проставим на всех дугах веса. Вес c_{ij} показывает, какие затраты будет иметь фирма в годы, начиная с i и кончая c_{j-1} , если станок покупается в начале i -го года и продается к началу j -го года. Например,

$$c_{18}=p_1+r_1+r_2+r_3+r_4+r_5+r_6+r_7-v_7=100+30+40+50+50+60+70+100-0=500.$$

Аналогично:

$$c_{12}=100+30-50=80;$$

$$c_{13}=100+30+40-25=145;$$

$$c_{14}=100+30+40+50-10=210;$$

$$c_{15}=100+30+40+50+50-5=265;$$

$$c_{16}=100+30+40+50+50+60-2=328;$$

$$c_{17}=100+30+40+50+50+60+70-1=399;$$

$$c_{23}=105+30-50=85;$$

$$c_{24}=105+30+40-25=150;$$

$$c_{25}=105+30+40+50-10=215;$$

$$c_{26}=105+30+40+50+50-5=270;$$

$$c_{27}=105+30+40+50+50+60-2=333;$$

$$c_{28}=105+30+40+50+50+60+70-1=404;$$

$$c_{34}=110+30-50=90;$$

$$c_{35}=110+30+40-25=155;$$

$$c_{36}=110+30+40+50-10=220;$$

$$c_{37}=110+30+40+50+50-5=275;$$

$$c_{38}=110+30+40+50+50+60-2=338;$$

$$c_{45}=115+30-50=95;$$

$$c_{46}=115+30+40-25=160;$$

$$c_{47}=115+30+40+50-10=225;$$

$$c_{48}=115+30+40+50+50-5=280;$$

$$c_{56}=120+30-50=100;$$

$$c_{57}=120+30+40-25=165;$$

$$c_{58}=120+30+40+50-10=230;$$

$$c_{67}=125+30-50=105;$$

$$c_{68}=125+30+40-25=170;$$

$$c_{78}=130+30-50=110.$$

Вообще говоря, можно произвести вычисления только для дуг, исходящих из узла 1. Дуги из других узлов отличаются от них только на величины, кратные 5.

Задача 4.4

Это задача на применение алгоритма определения кратчайшего пути. Пусть имеется ациклическая сеть (рис. 55). Требуется определить кратчайшие маршруты из всех узлов в узел 1 и найти их длину.

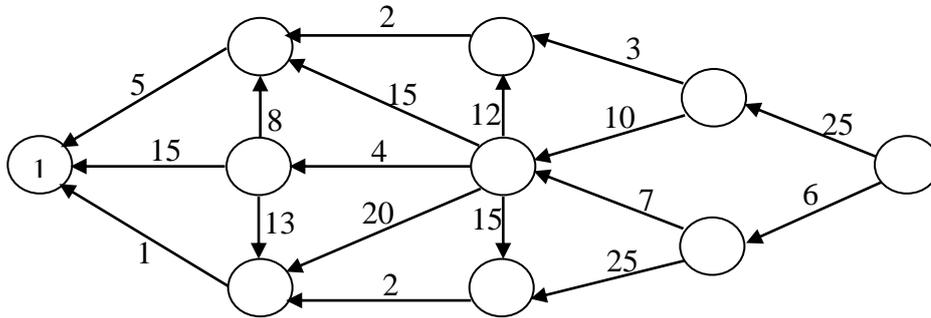


Рис. 55. Исходная ациклическая сеть для задачи 4.4

Решение

1. Выполним нумерацию узлов. Для этого зачеркнем все дуги, входящие в узел 1 и не будем их рассматривать в дальнейшем при присвоении номеров. Возьмем любой другой узел, имеющий теперь только входящие в него дуги, и припишем ему номер 2. Зачеркнем этот узел и все входящие в него дуги и исключим их из рассмотрения при дальнейшем присвоении номеров. Будем продолжать этот процесс до тех пор, пока все узлы не будут пронумерованы. На рис.56 изображена сеть с пронумерованными узлами.

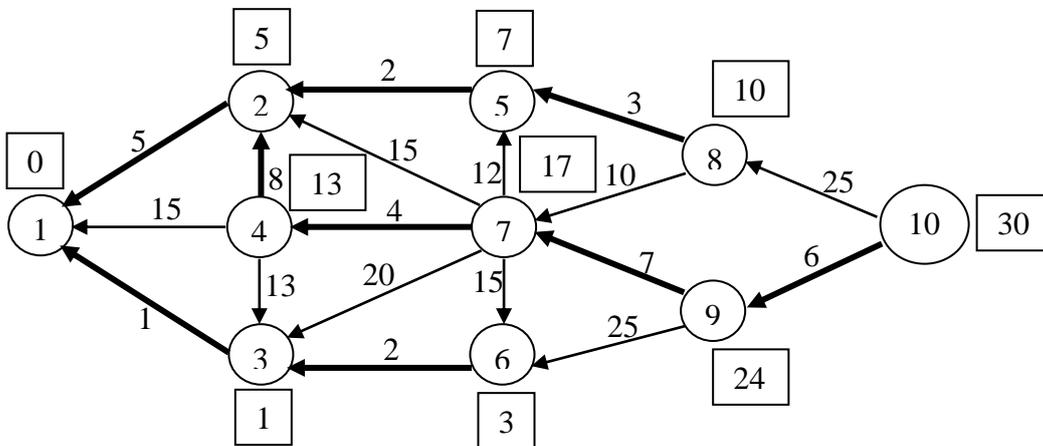


Рис. 56. Решение задачи поиска кратчайших путей в пункт 1

2. Находим кратчайшие расстояния от каждой вершины в вершину 1 по формулам:

$$y_1=0, \quad y_i = \min_{(i,j) \in \text{сети}} (c_{ij} + y_j) \quad \text{при } i = 2, 3, \dots, 10.$$

В результате:

$$y_1=0,$$

$$y_2=\min (5+0) = 5,$$

$$\begin{aligned}
y_3 &= \min (1+0) = 1, \\
y_4 &= \min (15+0, 8+5, 13+1) = 13, \\
y_5 &= \min (2+5) = 7, \\
y_6 &= \min (2+1) = 3, \\
y_7 &= \min (12+7, 15+5, 4+13, 20+1, 15+3) = 17, \\
y_8 &= \min (3+7, 10+17) = 10, \\
y_9 &= \min (7+17, 25+3) = 24, \\
y_{10} &= \min (25+10, 6+24) = 30.
\end{aligned}$$

Длины кратчайших путей поместим в прямоугольники около вершин, а сами кратчайшие пути выделим жирными линиями (рис. 56).

Задача 4.5

Фирма должна выполнить обязательства по поставке своей продукции в течение периодов 1,2,3,4 в объеме соответственно 12,9,18,22 единиц.

При условии односменной работы объем выпуска за один период составляет соответственно 10,8,10,12 единиц. При этом затраты на производство одной единицы продукции составляют 10 тыс. рублей. Пользуясь сверхурочными работами фирма может выполнить в зависимости от периода дополнительно 8,6,5,4 единиц продукции при затратах на производство каждой единицы в размере 15 тыс. рублей. Предприятие может создавать запасы с целью удовлетворения спроса в последующие периоды. Хранение единицы продукции в течение одного периода составляет 1 тыс. рублей.

Требуется построить сеть.

Решение

На рис. 57 приведена сеть классической транспортной задачи, отражающая деятельность фирмы. Заметим, что суммарное «предложение» при этом превышает суммарный «спрос», поэтому требуется введение фиктивного «потребителя». Пользуясь этой сетью нетрудно выполнить запись математической модели, а также построить матрицу транспортной задачи.

Задача 4.6

Фирма добывает руду на двух рудниках 1 и 2, перевозит ее автомобилями в пункт 3, где находится железнодорожная станция, и в пункт 4, где находится речной порт. Из пунктов 3 и 4 руда перевозится в пункты потребления 5,6,7.

Перевозка одной тонны руды по автомобильной дороге обходится в 0,5 единиц денег, по железной дороге 0,3 единицы, по воде – 0,2 единицы на каждый километр. Расстояния между пунктами изображены на рис.58. Мощность рудников 1000 тонн и 2000 тонн в день, ежедневный спрос – 500, 1500, 800.

Требуется построить матрицу классической транспортной задачи.

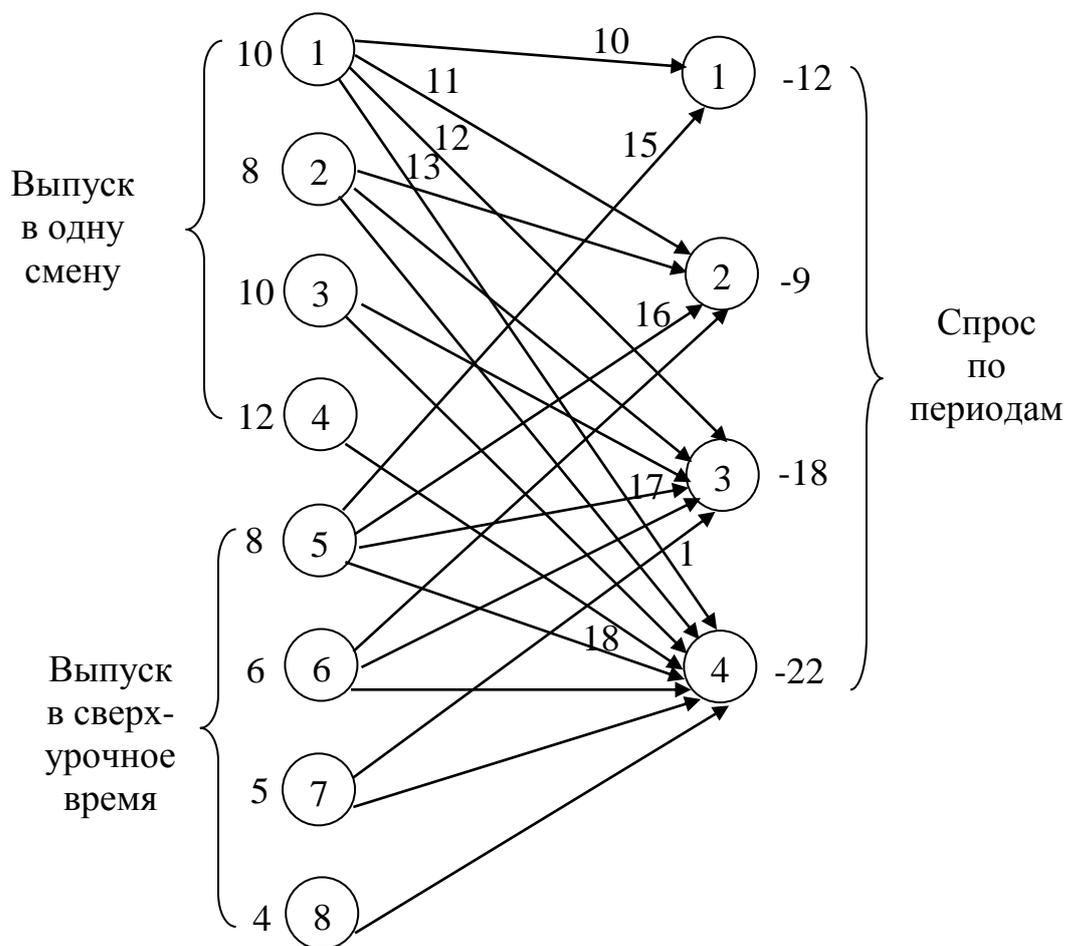


Рис. 57. Сеть для задачи 4.5

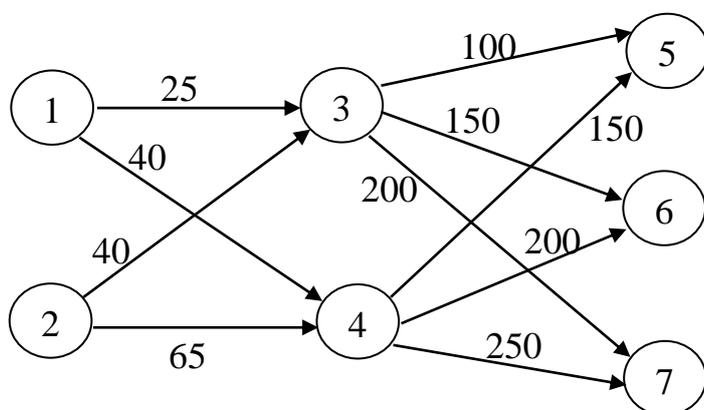


Рис. 58. Карта перевозок для задачи 4.6

Решение

Во-первых, построим сеть транспортной задачи с промежуточными пунктами (рис. 59). Веса дуг имеют смысл затрат, обусловленных перевозкой по соответствующей дороге единицы груза.

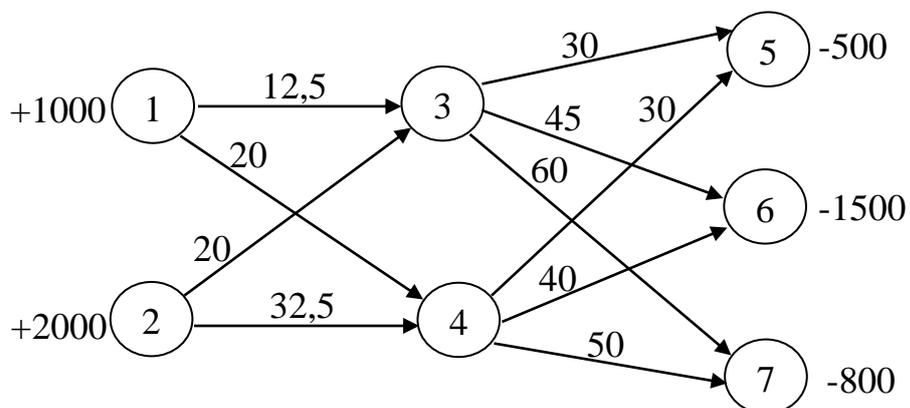


Рис. 59. Сеть транспортной задачи с промежуточными пунктами

Во-вторых, найдем кратчайший путь из каждого рудника к каждому из пунктов потребления:

$$c_{15}=42,5, c_{16}=57,5, c_{17}=70,$$

$$c_{25}=50, c_{26}=65, c_{27}=80.$$

В-третьих, введя фиктивного потребителя со спросом 200, запишем матрицу условий соответствующей классической транспортной задачи (рис. 60).

ПО \ ПН	5	6	7	8	Поставки
1	42,5	57,5	70	0	1000
2	50	65	80	0	2000
Спрос	500	1500	800	200	3000

Рис. 60. Матрица условий классической транспортной задачи

5. ЦЕЛОЧИСЛЕННОЕ ПРОГРАММИРОВАНИЕ

5.1 Общее описание модели

Рассмотрим следующую модель:

$$\sum_{j=1}^n c_j x_j \Rightarrow \max, \quad (5.1)$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad (i = \overline{1, m}), \quad (5.2)$$

$$x_j \geq 0 \quad (j = \overline{1, n}), \quad (5.3)$$

$$x_j - \text{целые} \quad (j = \overline{1, p}, p \leq n). \quad (5.4)$$

Задача типа (5.1) – (5.4) называется задачей *целочисленного (дискретного) программирования*. Если $p=n$, то есть все переменные должны быть целыми числами, то модель является полностью целочисленной. Если $p < n$ имеем частично целочисленную модель. Модель (5.1)–(5.4) представлена в канонической форме, к которой может быть приведена любая задача целочисленного программирования, пользуясь теми же приемами, что и для преобразования задач линейного программирования (см. п.3.2).

Задача целочисленного программирования (5.1)–(5.4) отличается от задачи линейного программирования наличием условия целочисленности (5.4). Принятие данного условия означает невыполнение гипотезы делимости, принимаемой при построении моделей линейного программирования. В случае, когда линейные ограничения (5.2) соответствуют некоторой сетевой модели, существует оптимальное решение задачи (5.1)–(5.3), удовлетворяющее условию целочисленности (5.4). Однако в общем случае условие (5.4) приводит к тому, что максимальное значение целевой функции задачи целочисленного программирования оказывается меньше, чем значение целевой функции соответствующей задачи линейного программирования (5.1) – (5.3).

Примером содержательной постановки задачи (5.1) – (5.4) является задача определения оптимального плана выпуска продукции. В этой содержательной постановке, которая ничем не отличается от постановок задач линейного программирования, некоторые из переменных x_j описывают небольшие объемы выпуска каких-то дорогостоящих изделий, например, прокатных станов, океанских танкеров и т.д. Для таких изделий дробное количество, например $\frac{10}{3}$, как правило, не имеет смысла.

Ниже описываются более сложные модели целочисленного программирования, некоторые из переменных которых имеют содержательную интерпретацию, не встречающуюся в моделях линейного программирования.

5.2. Примеры моделей целочисленного программирования

5.2.1. Оптимальный выбор набора проектов для реализации

Подобные модели, в которых осуществляется выбор альтернатив «да – нет», являются наиболее важными моделями целочисленного программирования.

Пусть имеется n проектов, которые могут быть реализованы. Проекты могут быть самые разнообразные: возведения предприятий, разработки новых видов изделий, по обеспечению сохранности природной среды. Так как имеющиеся ресурсы ограничены, имеется возможность выполнить только часть из n проектов.

Введем управляемые переменные:

$$x_j = \begin{cases} 1, & \text{если проект } j \text{ будет выполнен,} \\ 0, & \text{не будет выполнен.} \end{cases}$$

Ограничения, описывающие потребление ресурсов, имеют вид:

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

где a_{ij} – объем ресурса i , идущего на реализацию проекта j .

Кроме ограничений на потребление ресурсов, которые аналогичны ограничениям задачи линейного программирования, двоичный (булевый) характер управляемых переменных позволяет описать различные требования, предъявляемые к выбираемому набору проектов. Например, пусть известно, что из первых p проектов не может быть реализовано более k проектов, так как для выполнения каждого проекта требуется наличие ведущего инженера, а имеется возможность выделить всего k инженеров такой квалификации. Тогда это условие можно выразить с помощью линейного ограничения:

$$\sum_{j=1}^p x_j \leq k.$$

Если записать последнее ограничение в виде равенства, то будут приняты к исполнению точно k проектов. Особый случай наличия p взаимоисключающих (альтернативных) проектов можно учесть, приняв $k=1$.

Если проект 2 может быть принят лишь при условии, что принят проект 1, то это условие можно задать следующим ограничением: $-x_1 + x_2 \leq 0$.

Допустим, известно, что проекты 1 и 2 являются взаимоисключающими и проект 3 можно выполнить только при условии, что принят или 1 или 2. Данную зависимость можно описать одновременным заданием двух ограничений:

$$\begin{aligned} x_1 + x_2 &\leq 1, \\ -x_1 - x_2 + x_3 &\leq 0. \end{aligned}$$

Теперь уже очевидно, что самые разнообразные комбинаторные зависимости между проектами можно задать с помощью линейных ограничений, накладываемых на переменные x_j .

5.2.2. Задача размещения предприятий

Данная модель может рассматриваться как развитие классической транспортной модели, которая описывает снабжение n потребителей определенного вида продукции m имеющимися поставщиками этой продукции. В данной задаче также n потребителей и m поставщиков. Но в отличие от классической транспортной задачи, поставщики, то есть соответствующие заводы или склады, еще не возведены. Поэтому в модели должны быть учтены затраты на их создание. Кроме того, пока еще нет некоторых дорог от пунктов производства к пунктам потребления. Эти дороги можно построить, если вложить соответствующие средства.

Построение математической модели начнем с введения целочисленных управляемых переменных:

$$y_i = \begin{cases} 1, & \text{если будет построен завод - поставщик } i \quad (i = \overline{1, m}), \\ 0, & \text{не будет,} \end{cases}$$

$$z_{ij} = \begin{cases} 1, & \text{если будет построена дорога из } i \text{ в } j \quad (i = \overline{1, m}, j = \overline{1, n}), \\ 0, & \text{не будет,} \end{cases}$$

x_{ij} ($i = \overline{1, m}, j = \overline{1, n}$) - количество продукции, которое будет поставлено поставщиком i потребителю j .

Математическая модель имеет вид:

$$\sum_{i=1}^m f_i y_i + \sum_{i=1}^m \sum_{j=1}^n f_{ij} z_{ij} + \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \Rightarrow \min, \quad (5.5)$$

$$\sum_{j=1}^n x_{ij} - S_i y_i \leq 0 \quad (i = \overline{1, m}), \quad (5.6)$$

$$\sum_{i=1}^m x_{ij} = D_j \quad (j = \overline{1, n}), \quad (5.7)$$

$$x_{ij} - U_{ij} z_{ij} \leq 0 \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (5.8)$$

$$x_{ij} \geq 0 \quad (i = \overline{1, m}, j = \overline{1, n}), \quad (5.9)$$

$$y_i, z_{ij} = 0, 1 \quad (i = \overline{1, m}, j = \overline{1, n}). \quad (5.10)$$

В (5.5) коэффициент f_i ($i = \overline{1, m}$) есть затраты на возведение поставщика i , f_{ij} есть стоимость строительства дороги из i в j , c_{ij} - затраты на изготовление одной единицы продукции на заводе i и доставку ее в пункт j . Каждое из

ограничений (5.6) отражает ограниченность поставок соответствующего поставщика. Ограничения (5.7) описывают спрос. В (5.8) коэффициенты U_{ij} есть достаточно большие числа. Например, можно взять $U_{ij} = \min(S_i, D_j)$. Каждое из данных ограничений задает невозможность перевозки x_{ij} , если соответствующая дорога не будет построена.

Так как возведение того или иного предприятия, или дороги может рассматриваться как реализация соответствующего проекта, модель (5.5) – (5.10) может быть расширена путем добавления ограничений, присущих предыдущей модели. Например, условие, чтобы из пункта 3 не выходило более 3-х дорог, можно задать следующим ограничением: $\sum_{j=1}^n Z_{3j} \leq 3$.

5.2.3. Распределение операций по исполнителям

Рассматриваемые ранее модели целочисленного программирования описывают задачи стратегического планирования. Для этих задач характерны большие длительности периодов планирования, измеряемые годами и большие объемы распределяемых ресурсов, иногда на миллионы рублей. Следующие модели являются моделями оперативного планирования. Единицы измерения времени и ресурсов для этих задач гораздо мельче.

Имеется m операций и n исполнителей. В отличие от задачи о назначениях некоторые из исполнителей могут выполнить в течение планового периода не одну, а несколько операций.

Введем управляемые переменные и запишем целевую функцию аналогично задаче о назначениях:

$$x_{ij} = \begin{cases} 1, & \text{работа } i \text{ выполняется исполнителем } j \\ 0, & \text{не выполняется,} \end{cases}$$

$$(i = \overline{1, m}, j = \overline{1, n}),$$

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

где c_{ij} – стоимость выполнения операции i исполнителем j .

В отличие от задачи о назначениях, в данной модели может быть учтена ограниченность того или иного ресурса:

$$\sum_{i=1}^m \sum_{j=1}^n a_{kij} x_{ij} \leq b_k \quad (k = \overline{1, K}),$$

где a_{kij} – затраты k -го ресурса при выполнении работы i исполнителем j .

Подобно задаче о назначениях, каждая работа выполняется строго один раз:

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

Если исполнитель 1 не может выполнить более двух работ, это можно задать ограничением:

$$\sum_{i=1}^m x_{i1} \leq 2.$$

Аналогично задаче выбора проектов, могут быть заданы любые другие зависимости между x_{ij} .

5.2.4. Задача составления расписания

Предположим, что имеется m деталей, которые нужно обработать на n станках. Каждая деталь должна сначала обрабатываться на станке 1, затем на станке 2 и т.д. до n . Время обработки детали i на станке j равно t_{ij} . Требуется определить порядок обработки деталей на каждом из станков такой, чтобы минимизировать время, нужное для изготовления всех m деталей.

Введем управляемые переменные: x_{ij} ($i = \overline{1, m}, j = \overline{1, n}$) – момент времени, в который деталь i начинает обрабатываться на станке j . Время отсчитывается от начала обработки партии из m деталей.

Нетрудно заметить, что обработка детали i закончится в момент времени: $x_{in} + t_{in}$ ($i = \overline{1, m}$). Последней закончит обрабатываться деталь, которой соответствует: $\max_i (x_{in} + t_{in})$. Отсюда можно записать целевую функцию:

$$\max_i (x_{in} + t_{in}) \Rightarrow \min. \quad (5.11)$$

Выражение (5.11) эквивалентно совокупности линейной целевой функции (5.12) и m линейных ограничений (5.13):

$$x_0 \Rightarrow \min, \quad (5.12)$$

$$x_0 \geq x_{in} + t_{in} \quad (i = \overline{1, m}). \quad (5.13)$$

С помощью следующей группы ограничений реализуем требование о том, что в любой момент времени на каждом станке не может обрабатываться более одной детали. Рассмотрим некоторый станок j . Определенная деталь i обрабатывается на нем с момента x_{ij} до момента $x_{ij} + t_{ij}$. Другая деталь k обрабатывается с момента x_{kj} до $x_{kj} + t_{kj}$. Так как данные интервалы времени не должны пересекаться, обязательно должно выполняться условие:

$$x_{ij} + t_{ij} \leq x_{kj} \quad (5.14)$$

или условие:

$$x_{ij} \geq x_{kj} + t_{kj}. \quad (5.15)$$

В линейном программировании нельзя задавать ограничения «или-или», но это можно сделать в целочисленном программировании. Для этого введем управляемые переменные:

$$y_{ikj} (i = \overline{1, m-1}, k = \overline{i+1, m}, j = \overline{1, n}) = \begin{cases} 1, & \text{если работа } i \text{ выполняется на станке } j \\ & \text{раньше, чем работа } k, \\ 0, & \text{если наоборот.} \end{cases}$$

С учетом введенных переменных ограничения, запрещающие одновременную обработку двух деталей на одном станке, примут вид:

$$\begin{aligned} x_{ij} - x_{kj} + (t_{ij} + V)y_{ikj} &\leq V, \\ x_{ij} - x_{kj} - t_{kj} &\geq -Vy_{ikj}, \end{aligned} \quad (5.16)$$

$$i = \overline{1, m-1}, k = \overline{i+1, m}, j = \overline{1, n}.$$

где V – достаточно большое положительное число, например, в качестве V можно взять величину, соизмеримую со временем изготовления всей партии деталей.

Нетрудно убедиться, что если задать $y_{ikj}=0$, то первое ограничение в (5.16) выполняется всегда, а второе совпадает с (5.15). При $y_{ikj}=1$ всегда выполняется второе ограничение, а первое совпадает с (5.14).

Последняя группа ограничений задает правильный порядок обработки одной и той же детали на различных станках:

$$x_{i,j-1} + t_{i,j-1} \leq x_{ij} \quad (i = \overline{1, m}, j = \overline{2, n}). \quad (5.17)$$

То есть до тех пор, пока деталь i не закончит обрабатываться на станке $j-1$, она не может начать обрабатываться на станке j .

Таким образом, оптимальное расписание можно найти, решив частично целочисленную задачу (5.12) - (5.13), (5.16) - (5.17), включающую также ограничения (5.18):

$$x_{ij} \geq 0 \quad (i = \overline{1, m}, j = \overline{1, n}), \quad y_{ikj} = 0, 1 \quad (i = \overline{1, m-1}, k = \overline{i+1, m}; j = \overline{1, n}). \quad (5.18)$$

5.2.5. Задача с альтернативными ограничениями

В предыдущей задаче рассмотрен случай двух альтернативных ограничений, рассмотрим еще один подобный пример. Пусть x_j есть количество изделий, которое будет выпускаться в течение некоторого планового периода. При этом возможны две ситуации:

- а) $x_j \geq L_j$, где L_j – наименьший допустимый размер партии;
- б) $x_j = 0$.

Тогда данную ситуацию можно выразить, введя булеву переменную $y_j=0,1$ и два линейных ограничения:

$$x_j - U_j y_j \leq 0, \quad (5.19)$$

$$x_j - L_j y_j \geq 0, \quad (5.20)$$

где U_j – достаточно большое положительное число, заведомо превышающее объем выпуска.

При $y_j=0$ из (5.19) и (5.20) следует, что изделия не выпускаются, т.е. $x_j=0$. При $y_j=1$ ограничение (5.19) можно не учитывать, а из (5.20) вытекает требование к размеру партии.

Изложенный подход можно обобщить, распространив его на случай, когда решение должно удовлетворять по крайней мере k из p ограничений:

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, 2, \dots, p.$$

Введем p булевых переменных $y_i = 0, 1$ ($i = \overline{1, p}$) и наложим $1+p$ линейных ограничений:

$$\begin{aligned} \sum_{i=1}^p y_i &\geq k, \\ \sum_{j=1}^n a_{ij} x_j &\leq b_i y_i + U_i (1 - y_i), \quad (i = \overline{1, p}), \\ \text{или} \quad \sum_{j=1}^n a_{ij} x_j - (b_i - U_i) y_i &\leq U_i, \quad (i = \overline{1, p}), \end{aligned}$$

где значение U_i выбирается настолько большим, чтобы условие $\sum_{j=1}^n a_{ij} x_j \leq U_i$ обязательно выполнялось бы при оптимальном решении.

5.3 Решение задачи целочисленного программирования

Для поиска оптимального решения задачи целочисленного программирования (5.1)–(5.4) будем использовать одну из модификаций метода ветвей и границ, рассмотренного в п.4.5.2. Данный алгоритм можно применять для решения как полностью, так и частично целочисленных задач. Данный алгоритм включает следующие шаги.

Шаг 1. Исходная задача заносится в дерево задач. Берется любое допустимое решение задачи. Соответствующее ему значение целевой функции берется в качестве нижней (худшей) границы x_{OH}^1 (1 – номер итерации) оптимального значения целевой функции.

Шаг 2. Прекратить вычисления, если нерассмотренных ранее висячих вершин дерева задач больше нет. Иначе – выбрать одну из данных задач.

Шаг 3. Решить соответствующую выбранной задаче задачу линейного программирования. Полученное значение целевой функции записать в качестве верхней (лучшей) границы x_{OB} целевой функции целочисленной задачи. Подобное получение верхней границы обусловлено тем, что задача линейного программирования по сравнению с соответствующей задачей целочисленного

программирования не имеет ограничений целочисленности переменных (5.4).

Если $x_{OB} \leq x_{OH}^t$, то принять $x_{OH}^{t+1} = x_{OH}^t$ и вернуться к шагу 2. Иначе – переход на шаг 4.

Шаг 4. Если полученное оптимальное решение задачи линейного программирования удовлетворяет целочисленным ограничениям, то зафиксировать его в качестве нижней границы x_{OH}^{t+1} целевой функции и вернуться на шаг 2. Иначе переход на шаг 5.

Шаг 5. Выбрать любую переменную $x_j (j = \overline{1, p})$, которая должна быть целочисленной, но не является таковой в полученном решении задачи линейного программирования. Пусть d_j есть значение этой переменной, а $[d_j]$ определяет наибольшее число, меньшее d_j . Внести в дерево задач в качестве «дочерних» задачи, выбранной на шаге 2, две задачи, одна из которых отличается от задачи, полученной на шаге 2, наличием добавочного ограничения $x_j \leq [d_j]$, а вторая: $x_j \geq [d_j] + 1$.

Принять $x_{OH}^{t+1} = x_{OH}^t$ и вернуться на шаг 2.

Пример. Требуется найти решение следующей модели:

$$\begin{aligned} 3x_1 + 3x_2 + 13x_3 &\Rightarrow \max, \\ -3x_1 + 6x_2 + 7x_3 &\leq 8, \\ 6x_1 - 3x_2 + 7x_3 &\leq 8, \\ x_j &= 0, 1, 2, \dots (j = 1, 2, 3). \end{aligned}$$

Шаг 1. Исходную задачу 1 заносим в дерево задач (рис.61).

В качестве исходного допустимого решения берем: $x_1 = x_2 = x_3 = 0$. Соответствующее значение целевой функции берем в качестве нижней границы оптимального значения целевой функции: $x_{OH}^1 = 0$.

Итерация 1

Шаг 2. Выбираем единственную висячую вершину дерева задач – задачу 1.

Шаг 3. Решаем задачу линейного программирования, соответствующую задаче 1. Полученное решение имеет вид: $x_1 = x_2 = 2\frac{2}{3}$, $x_3 = 0$, $x_{OB} = 16$.

Шаг 4. Полученное оптимальное решение задачи линейного программирования не является целочисленным, поэтому переход на шаг 5.

Шаг 5. Выбираем переменную x_1 . Запишем в дерево задач две новые задачи – 2 и 3. Первая из них отличается от задачи 1 тем, что в ней добавляется новое ограничение $x_1 \geq 3$, а вторая: $x_1 \leq 2$. Принимаем $x_{OH}^2 = x_{OH}^1 = 0$.

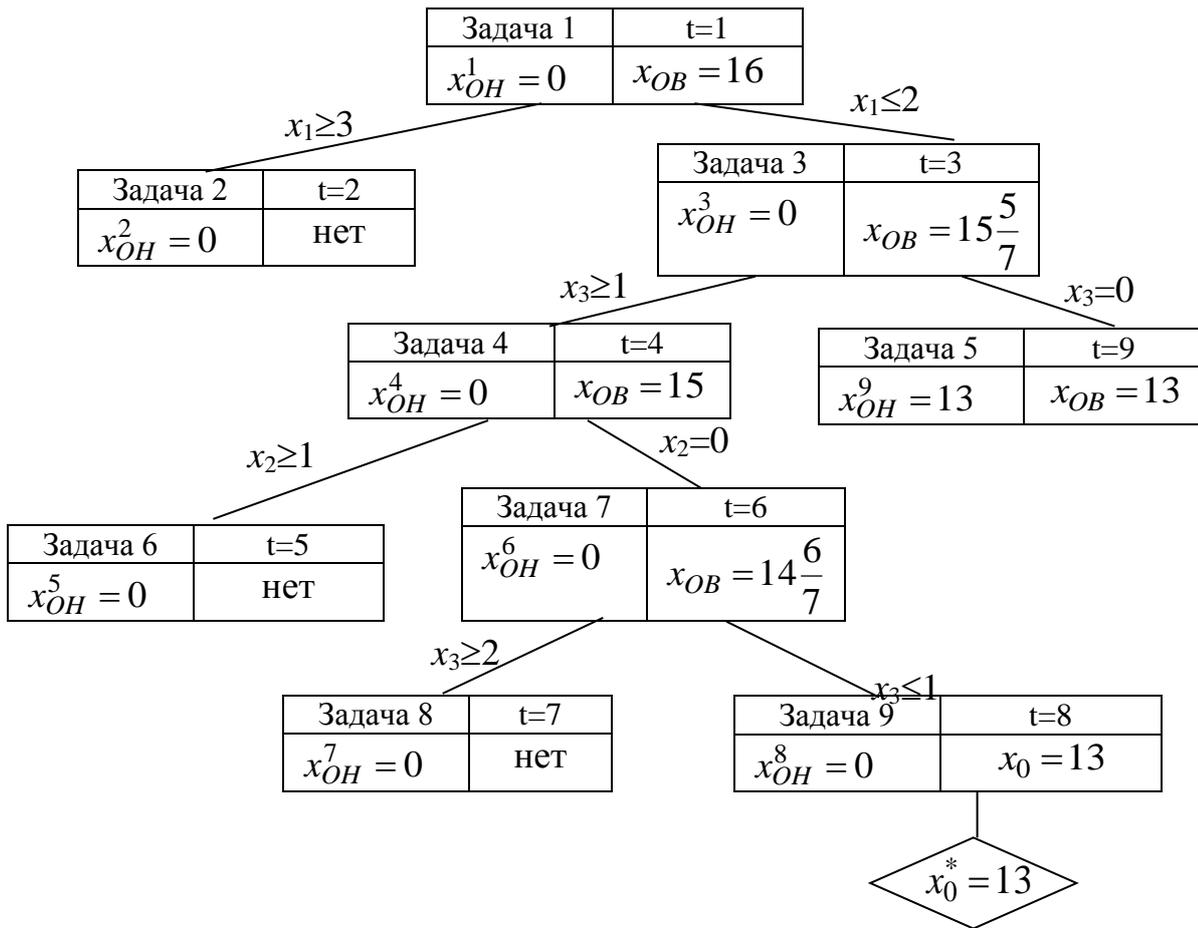


Рис. 61. Дерево задач в примере решения целочисленной модели

Итерация 2

Шаг 2. Выбираем задачу 2.

Шаг 3. Задача линейного программирования, соответствующая задаче 2, не имеет допустимого решения. Принимаем $x_{OH}^3 = x_{OH}^2 = 0$. Переход на шаг 2.

Итерация 3

Шаг 2. Выбираем задачу 3.

Шаг 3. Находим оптимальное решение задачи линейного программирования, соответствующей задаче 3:

$$x_1 = x_2 = 2, \quad x_3 = \frac{2}{7}, \quad x_{OB} = 15\frac{5}{7}.$$

Шаг 4. Решение не является целочисленным, поэтому переход на шаг 5.

Шаг 5. Выберем x_3 . Внесем в дерево задач задачу 4 с добавочным ограничением $x_3 \geq 1$ и задачу 5 с ограничением $x_3 = 0$. Принимаем $x_{OH}^4 = x_{OH}^3 = 0$. Переход на шаг 2.

Итерация 4

Шаг 2. Выбираем задачу 4.

Шаг 3. Оптимальное решение задачи линейного программирования:

$$x_1 = x_2 = \frac{1}{3}, x_3 = 1, x_{OB} = 15.$$

Шаг 4. Решение не является целочисленным, поэтому переход на шаг 5.

Шаг 5. Выберем x_2 . Вносим в дерево задач задачу 6 с добавочным ограничением $x_2 \geq 1$, задачу 7 с ограничением $x_2 = 0$. $x_{OH}^5 = x_{OH}^4 = 0$.

Итерация 5

Шаг 2. Выбираем задачу 6.

Шаг 3. Задача линейного программирования не имеет допустимого решения. Принимаем $x_{OH}^6 = x_{OH}^5 = 0$.

Итерация 6

Шаг 2. Выбираем задачу 7.

Шаг 3. Оптимальное решение задачи линейного программирования:

$$x_1 = x_2 = 0, x_3 = 1\frac{1}{7}, x_{OB} = 14\frac{6}{7}.$$

Шаг 4. Решение не является целочисленным, поэтому переход на шаг 5.

Шаг 5. Выберем x_3 . В дерево задач вносим задачу 8 с добавочным ограничением $x_3 \geq 2$ и задачу 9 с ограничением $x_3 \leq 1$. Принимаем $x_{OH}^7 = x_{OH}^6 = 0$.

Итерация 7

Шаг 2. Выбираем задачу 8.

Шаг 3. Задача линейного программирования не имеет допустимого решения. Принимаем $x_{OH}^8 = x_{OH}^7 = 0$.

Итерация 8

Шаг 2. Выбираем задачу 9.

Шаг 3. Решение задачи линейного программирования:

$$x_1 = x_2 = 0, x_3 = 1, x_0 = 13.$$

Шаг 4. Так как решение задачи линейного программирования целочисленно, принимаем $x_{OH}^9 = x_0 = 13$. Переход на шаг 2.

Итерация 9

Шаг 2. Выбираем задачу 5.

Шаг 3. Решение задачи линейного программирования:
 $x_1 = 2, x_2 = 2\frac{1}{3}, x_3 = 0, x_{OB} = 13$. Так как $x_{OB} \leq x_{OH}^9$, то принимаем
 $x_{OH}^{10} = x_{OH}^9 = 13$.

Итерация 10

Шаг 2. Так как нерассмотренных ранее висячих вершин дерева задач больше нет, то выполнение алгоритма закончено: $x_O^* = x_{OH}^{10} = 13$. Оптимальное решение: $x_1^* = 0, x_2^* = 0, x_3^* = 1$.

5.4 Примеры задач

Задача 5.1

Студент решил прослушать лекции по пяти предметам – 1, 2, 3, 4, 5. Каждый предмет читают в четырех группах (1, 2, 3, 4), занимающихся в разное время. Студенту известно каждое время t_{ij} ($i = \overline{1,5}, j = \overline{1,4}$) – начало занятий по предмету i в группе j (рис. 62).

j	1					2					3					4				
i	1	2	3	4	5	5	4	3	2	1	2	1	4	5	3	3	4	1	5	2
t_{ij}	9	10	11	12	13	9	10	11	12	13	9	10	11	12	13	9	10	11	12	13

Рис. 62. Перечень t_{ij} (время начала занятия i в группе j)

В течение дня в каждой группе изучаются все 5 предметов. Время занятий – с 9 до 14 часов. Продолжительность одного занятия – 1 час. На выбор студента оказывает влияние репутация преподавателя. Пусть p_{ij} – полезность прослушивания предмета i в группе j .

Требуется построить модель для выбора расписания студента.

Решение

Введем управляемые переменные:

$$x_{ij} = \begin{cases} 1, & \text{предмет } i \text{ слушается студентом в группе } j, \\ 0, & \text{нет.} \end{cases}$$

Запишем математическую модель:

$$\sum_{i=1}^5 \sum_{j=1}^4 p_{ij} \cdot x_{ij} \Rightarrow \max, \quad (5.21)$$

$$\sum_{j=1}^4 x_{ij} = 1 \quad (i = \overline{1,5}), \quad (5.22)$$

$$\begin{aligned} x_{11} + x_{52} + x_{23} + x_{34} &= 1, \\ x_{21} + x_{42} + x_{13} + x_{44} &= 1, \\ x_{31} + x_{32} + x_{43} + x_{14} &= 1, \\ x_{41} + x_{22} + x_{53} + x_{54} &= 1, \\ x_{51} + x_{12} + x_{33} + x_{24} &= 1, \\ x_{ij} &= 0,1, \quad i = \overline{1,5}, \quad j = \overline{1,4}. \end{aligned} \quad (5.23)$$

$$x_{ij} = 0,1, \quad i = \overline{1,5}, \quad j = \overline{1,4}. \quad (5.24)$$

Каждое из ограничений (5.22) требует, чтобы соответствующий предмет изучался студентом в течение дня один и только один раз. Каждое из ограничений (5.23) требует, чтобы студент находился в течение часа только в одной аудитории. Например, первое из этих ограничений определяет поведение студента с 9 до 10 часов, второе – с 10 до 11 часов и т.д.

Задача 5.2

Фирма должна составить расписание предупредительного ремонта пяти своих агрегатов (1, 2, 3, 4 и 5), рассчитанного на восемь недель. Ремонт каждого агрегата длится три недели, в течение каждой из которых требуется определенное количество трудовых ресурсов. На рис.63 указаны эти количества, а также приведены раннее и позднее начало каждого ремонта. Например, для одной недели ремонта агрегата 1 требуется, соответственно, 4,6 или 3 человеко-недели. При этом ремонт может начаться в начале первой, второй, третьей или четвертой недели.

Агрегат	Первая неделя ремонта	Вторая неделя ремонта	Третья неделя ремонта	Раннее начало	Позднее начало
1	4	6	3	1	4
2	3	2	5	1	3
3	7	1	1	2	5
4	1	3	6	2	6
5	8	9	2	3	5

Рис. 63. Данные задачи 5.2 (ремонт агрегатов)

Предположим, что ЛПР хочет минимизировать максимальное количество трудовых ресурсов, используемых в течение любой недели. Требуется построить соответствующую математическую модель.

Решение

Введем управляемые переменные:

$$x_{ij} = \begin{cases} 1, & \text{если агрегат } i \text{ начал ремонтироваться в начале недели } j, \\ 0, & \text{если иначе,} \end{cases}$$

$$i = \overline{1,5}, j = \overline{1,6}.$$

В соответствии с условиями задачи (см. рис.63) некоторые из переменных x_{ij} тождественно равны нулю. Перечислим те x_{ij} , которые могут быть ненулевыми:

$$x_{11}, x_{12}, x_{13}, x_{14},$$

$$x_{21}, x_{22}, x_{23},$$

$$x_{32}, x_{33}, x_{34}, x_{35},$$

$$x_{42}, x_{43}, x_{44}, x_{45}, x_{46}$$

$$x_{53}, x_{54}, x_{55}.$$

Кроме того, введем переменную y – максимальные недельные трудозатраты.

Целевая функция модели:

$$y \Rightarrow \min.$$

Первая группа ограничений включает восемь ограничений, каждое из которых требует, чтобы y был не меньше, чем трудозатраты на соответствующей неделе. Первые три ограничения имеют вид:

$$4x_{11} + 3x_{21} \leq y,$$

$$6x_{11} + 4x_{12} + 2x_{21} + 3x_{22} + 7x_{32} + x_{42} \leq y,$$

$$3x_{11} + 6x_{12} + 4x_{13} + 5x_{21} + 2x_{22} + 3x_{23} + 1x_{32} + 7x_{33} + 3x_{42} + x_{43} + 8x_{53} \leq y.$$

Вторая группа ограничений включает пять ограничений, каждое из которых требует, чтобы соответствующий агрегат ремонтировался строго один раз:

$$x_{11} + x_{12} + x_{13} + x_{14} = 1,$$

$$x_{21} + x_{22} + x_{23} = 1,$$

$$x_{32} + x_{33} + x_{34} + x_{35} = 1,$$

$$x_{42} + x_{43} + x_{44} + x_{45} + x_{46} = 1,$$

$$x_{53} + x_{54} + x_{55} = 1.$$

Последние ограничения определяются смыслом переменных:

$$x_{ij} = 0, 1 (i = \overline{1,5}, j = \overline{1,6}).$$

Задача 5.3

Сборочная линия состоит из рабочих мест 1, 2, ..., на каждом из которых можно выполнить любое число операций по сборке изделия. Сборка каждого изделия включает шесть операций, продолжительности которых t_i приведены на рис. 64. Там же для каждой операции перечислены предшествующие операции, которые должны быть завершены прежде, чем начнет выполняться данная операция.

Операция i	Предшествующие операции	Время t_i
1	-	5
2	-	6
3	1	2
4	2, 3	5
5	1, 2	4
6	4	3

Рис. 64. Данные задачи 5.3 (планирование сборочной линии)

Суммарное время выполнения всех операций на одном рабочем месте не должно превышать величины T , называемой продолжительностью цикла линии. Пусть $T=10$.

Требуется построить математическую модель, решение которой позволит минимизировать число рабочих мест на линии.

Решение

Введем управляемые переменные:

$$x_{ij} = \begin{cases} 1, & \text{если операция } i \text{ выполняется на рабочем месте } j, \\ 0, & \text{если иначе,} \end{cases}$$

$$i = \overline{1,6}, j = \overline{1,6},$$

$$y_j = \begin{cases} 1, & \text{если рабочее место } j \text{ используется,} \\ 0, & \text{не используется,} \end{cases}$$

$$j = \overline{1,6}.$$

Целевая функция модели:

$$y_1 + y_2 + y_3 + y_4 + y_5 + y_6 \Rightarrow \min.$$

Первая группа из шести ограничений требует, чтобы операции выполнялись на рабочем месте только тогда, когда оно используется:

$$x_{11} + x_{21} + x_{31} + x_{41} + x_{51} + x_{61} \leq by_1,$$

.....

$$x_{16} + x_{26} + x_{36} + x_{46} + x_{56} + x_{66} \leq by_6.$$

Вторая группа ограничений отражает требования предшествования операций. Например, следующие ограничения отражают требование предшествования операции 1 по отношению к операции 3:

$$x_{31} \leq x_{11},$$

$$x_{32} \leq x_{11} + x_{12},$$

...

$$x_{36} \leq x_{11} + x_{12} + x_{13} + x_{14} + x_{15} + x_{16}.$$

Нетрудно подсчитать, что всего в этой группе будет: $4 \times 6 = 24$ ограничений.

Третья группа из шести ограничений обеспечивает, чтобы на любом рабочем месте суммарное время не превышало продолжительности цикла линии:

$$5x_{11} + 6x_{21} + 2x_{31} + 5x_{41} + 4x_{51} + 3x_{61} \leq 10,$$

.....

$$5x_{16} + 6x_{26} + 2x_{36} + 5x_{46} + 4x_{56} + 3x_{66} \leq 10.$$

Четвертая группа из пяти ограничений обеспечивает непрерывность последовательности используемых рабочих мест:

$$y_2 \leq y_1, \quad y_3 \leq y_2, \quad y_4 \leq y_3, \quad y_5 \leq y_4, \quad y_6 \leq y_5.$$

Таким образом, например, исключается ситуация, когда рабочее место 5 используется, а место 4 – нет.

Последняя группа ограничений:

$$x_{ij} = 0,1 \quad (i = \overline{1,6}, \quad j = \overline{1,6}), \quad y_j = 0,1 \quad (j = \overline{1,6}).$$

5.5. Заключение

Из приведенных выше примеров видно, что выразительные возможности моделей целочисленного программирования гораздо выше по сравнению с линейным программированием. То есть, пользуясь им можно описывать гораздо более сложные задачи принятия решений.

Тем не менее, линейное программирование имеет гораздо большее применение. Это обусловлено тем, что затраты на получение оптимального решения задачи целочисленного программирования, как правило, очень велики. Как было показано в п. 5.3, решение целочисленной задачи приводит к многократному решению задачи линейного программирования не меньшей размерности.

С другой стороны, размерность многих задач целочисленного программирования очень велика. Например, в описанной выше модели поиска расписания количество управляемых переменных равно $m \cdot n + m(m-1)n/2$, количество ограничений типа (5.16) - $m(m-1) \cdot n$, типа (5.17) - $m \cdot (n-1)$. Поэтому при четырех станках и десяти деталях имеем переменных: $10 \cdot 4 + 10 \cdot 9 \cdot 4/2 = 220$, ограничений: $10 \cdot 9 \cdot 4 + 10 \cdot 3 = 390$.

Поэтому при решении многих задач оперативного управления применение методов целочисленного программирования неэффективно, так как выгода от использования оптимального решения часто ниже затрат на его получение. Наиболее целесообразно использование методов целочисленного программирования в стратегическом планировании разработок дорогостоящих проектов. Что касается задач по составлению расписаний, то они часто решаются эвристически, приближенными методами. Исключением являются задачи с двумя и тремя станками, для которых разработаны очень простые оптимальные алгоритмы.

6. ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

6.1. Общее описание метода

Рассмотрим требования к задачам, решаемым с помощью динамического программирования. Во-первых, оптимизируемая операция должна быть «многошаговой» («многоэтапной») операцией. Иногда разбиение операции на шаги производится естественно. Например, деятельность фирмы в течение ряда лет. Другие операции можно разбить на шаги искусственно. Например, процесс наведения ракеты на цель можно условно разбить на этапы, каждый из которых занимает какое-то время Δt .

Во-вторых, показатель эффективности выполнения всей операции W определяется суммой показателей эффективности выполнения всех шагов операции: $W = \sum_{i=1}^m W_i$, где W_i – эффективность выполнения i -го шага. Иными словами, критерий W является аддитивным.

Обозначим множество управляемых переменных в данной операции как x . Тогда x можно представить в виде совокупности управляемых переменных, каждая из которых соответствует своему шагу выполнения операции: $x = (x_1, x_2, \dots, x_m)$, где $x_i (i = \overline{1, m})$ в общем случае не числа, а векторы. Тогда задачу, решаемую методом динамического программирования, можно сформулировать следующим образом: найти такое управление $x^* = (x_1^*, x_2^*, \dots, x_m^*)$, при котором W обращается в минимум: $W = \sum_{i=1}^m W_i \Rightarrow \min$.

В качестве примера рассмотрим следующую задачу нахождения кратчайшего пути на ациклической сети. Пусть некий турист хочет добраться из пункта 1 в пункт 10, пройдя путь наименьшей длины, ориентируясь по карте (рис. 65).

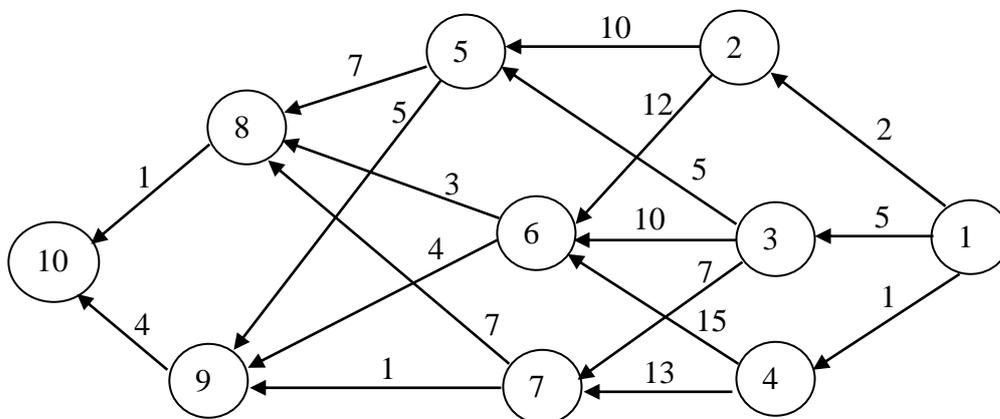


Рис. 65. Сеть (карта) возможного перемещения туриста

Разобьем движение туриста на шаги. При этом под одним шагом будем понимать переход туриста между двумя соседними пунктами. Нетрудно заметить, что какой бы из маршрутов из пункта 1 в пункт 10 мы не взяли, он включает 4 шага.

Показатель эффективности пройденного пути – длина этого пути. Она определяется суммой расстояний, пройденных на отдельных шагах. Следовательно, этот показатель является аддитивным. После того, как мы убедились, что исходная задача может быть решена методом динамического программирования, перейдем к построению соответствующей модели.

Сформулируем *принцип оптимальности*, играющий в динамическом программировании исключительно важную роль: независимо от того, как исследуемая операция достигла в прошлом некоторого своего состояния, последующие решения должны оптимизировать ее проведение, начиная с этого состояния. Например, для рассматриваемой задачи принцип оптимальности динамического программирования имеет вид: вне зависимости от того, как турист попал в прошлом в некоторый пункт j , дальнейший путь до пункта 10 он должен выбирать так, чтобы этот путь был самым коротким.

Изучив данный принцип оптимальности, турист понял, что оптимальный путь из пункта 6 не зависит от того, каким маршрутом он прибыл в него. Развивая эту идею далее, турист пришел к выводу, что если бы он знал оптимальные пути из пунктов 5, 6 и 7, то он легко бы нашел и оптимальный путь из пункта 3. Для этого достаточно просуммировать c_{35} , c_{36} или c_{37} с ранее вычисленной стоимостью оптимального пути из пунктов 5, 6 или 7, а затем выбрать тот пункт (5, 6 или 7), для которого эта сумма минимальна.

Для того чтобы математически описать принцип оптимальности, введем следующие обозначения:

$f_n(S)$ – длина кратчайшего пути от пункта (состояния) S , если до конечного пункта остается n шагов;

$x_n(S)$ – решение по шагу n (с конца), позволяющее достичь $f_n(S)$. Иными словами, $x_n(S)$ – номер пункта, в который мы должны перейти из пункта S , чтобы двигаться из S в конечный пункт по кратчайшему пути.

С учетом введенных обозначений опишем оптимальное поведение туриста. Обычно процесс динамического программирования разворачивается от конца к началу. То есть рассматривается последний шаг с конца, затем предпоследний и так далее до тех пор, пока не будет рассмотрен шаг 1.

n=0. Пусть турист уже находится в конечном пункте 10. Тогда справедливо:

$f_0(10)=0$, $x_0(10)$ =остановка, так как поход в пункте 10 заканчивается.

n=1. Пусть турист находится не в пункте 10, а на расстоянии одного перехода (этапа) от него. То есть он может находиться или в пункте 8 или в пункте 9. Тогда оптимальное поведение туриста зависит от пункта (состояния) и будет следующим:

$f_1(8) = c_{8,10} + f_0(10) = 1+0 = 1$, $x_1(8)=10$;

$f_1(9) = c_{9,10} + f_0(10) = 4+0 = 4$, $x_1(9)=10$.

Это можно представить в виде таблицы (рис. 66).

		$n=1$		
S	x	10	$x_1(S)$	$f_1(S)$
8		1+0	10	1
9		4+0	10	4

Рис. 66. Оптимальное поведение туриста на последнем этапе

$n=2$. Пусть турист находится на расстоянии двух этапов от пункта 10, то есть или в пункте 5, или в 6, или в 7. Тогда оптимальное дальнейшее поведение туриста будет следующим:

$$f_2(5) = \min[c_{5,8}+f_1(8), c_{5,9}+f_1(9)] = \min[7+1, 5+4] = 8, \quad x_2(5)=8;$$

$$f_2(6) = \min[c_{6,8}+f_1(8), c_{6,9}+f_1(9)] = \min[3+1, 4+4] = 4, \quad x_2(6)=8;$$

$$f_2(7) = \min[c_{7,8}+f_1(8), c_{7,9}+f_1(9)] = \min[7+1, 1+4] = 5, \quad x_2(7)=9.$$

Результаты вычислений на данном шаге удобно представить в виде таблицы (рис. 67).

		$n=2$			
S	x	8	9	$x_2(S)$	$f_2(S)$
5		7+1	5+4	8	8
6		3+1	4+4	8	4
7		7+1	1+4	9	5

Рис. 67. Оптимальное поведение туриста на предпоследнем этапе

$n=3$. Оптимальное поведение туриста на 3-м этапе от конечного пункта 10 приведено на рис. 68.

		$n=3$				
S	x	5	6	7	$x_3(S)$	$f_3(S)$
2		10+8	12+4	×	6	16
3		5+8	10+4	7+5	7	12
4		×	15+4	13+5	7	18

Рис. 68. Оптимальное поведение туриста на 3-м этапе от конца

$n=4$. За 4 этапа от конечного пункта турист может находиться только в одном состоянии - в пункте 1. Его оптимальное поведение при выходе из этого пункта показано на рис. 69.

		$n=4$			
$S \backslash x$	2	3	4	$x_4(S)$	$f_4(S)$
1	2+16	5+12	1+18	3	17

Рис. 69. Оптимальное поведение туриста на 4-м этапе от конца

Выполнявшиеся нами действия на любом шаге n ($n=1,2,3,4$) можно описать в виде одной формулы, называемой **рекуррентным соотношением динамического программирования**:

$$f_n(S) = \min_x [c_{Sx} + f_{n-1}(x)], \quad (S, x) \in \text{сети}. \quad (6.1)$$

Выражение (6.1) означает, что находясь на шаге n в состоянии S , для поиска оптимального управления необходимо рассмотреть все суммы дуг, выходящих из пункта S , с соответствующими оптимальными длинами путей, найденными на шаге $n-1$. Из указанных сумм необходимо выбрать наименьшую. Таким образом, для нахождения оптимальной стратегии на последних n шагах достаточно найти лишь оптимальное значение управляемой переменной для n -го шага, учитывая найденную ранее оптимальную стратегию для $(n-1)$ шагов.

После того, как в результате обратного просмотра этапов исходной задачи мы последовательно вычислили $f_1(S), f_2(S), \dots, f_n(S)$, необходимо просмотреть этапы задачи в прямом порядке. Сначала рассмотрим первый этап (с конца последний). Так как состояние системы (в примере это пункт 1) в начале первого этапа известно, мы можем найти оптимальное значение управляемой переменной на этом шаге. Из таблицы $n=4$ видно, что оптимально $x_4(1)=3$. Таким образом, второй этап от начала мы начинаем из состояния (пункта) 3. Из таблицы $n=3$ видно, что оптимально $x_3(3)=7$. Далее $x_2(7)=9$, $x_1(9)=10$. Таким образом, оптимальный путь проходит через пункты 1,3,7,9,10. Длина этого пути равна $f_4(1)=17$.

6.2 Задача управления запасами

Рассматриваемая ниже модель динамического программирования играет в исследовании операций очень важную роль. Хотя рассматриваемая ситуация и является идеализированной, в модели все же учитывается много важных факторов, влияющих на выбор решений по управлению запасами.

Предприятие должно разработать план выпуска некоторого вида изделий в течение N единиц времени. Предполагается, что для каждого t -го единичного отрезка времени имеется точный прогноз спроса на выпускаемую продукцию – D_t . Время изготовления партии изделий настолько мало, что им можно пренебречь, поэтому продукция, изготовленная в течение отрезка t может быть использована для покрытия спроса в течение этого отрезка. Предприя-

тие может изготавливать продукцию в объеме, превышающем спрос в пределах этого отрезка и хранить излишки, используя их для удовлетворения спроса на последующих отрезках.

Требуется разработать такой план выпуска, чтобы суммарные затраты на производство и на содержание запасов были минимальные при условии полного удовлетворения спроса на продукцию.

Введем управляемые переменные:

x_t – выпуск продукции в течение отрезка t ;

i_t – уровень запасов на конец отрезка t .

Пусть затраты на отрезке t есть некоторая функция $C_t(x_t, i_t)$, тогда целевую функцию можно записать в виде:

$$\sum_{t=1}^N C_t(x_t, i_t) \Rightarrow \min. \quad (6.2)$$

На значения переменных x_t и i_t наложено несколько ограничений. Во-первых, предполагается целочисленность объемов выпуска и уровней запасов:

$$\begin{aligned} x_t &= 0, 1, 2, \dots \quad (t = \overline{1, N}), \\ i_t &= 0, 1, 2, \dots \quad (t = \overline{1, N}). \end{aligned} \quad (6.3)$$

Во-вторых, предполагается, что на конец отрезка N должен быть нулевой уровень запасов: $i_N = 0$.

В-третьих, на каждом отрезке должны быть выполнены балансовые ограничения:

$$i_{t-1} + x_t - i_t - D_t = 0 \quad (t = \overline{1, N}). \quad (6.4)$$

Ограничения (6.4) линейны, поэтому, если бы все функции $C_t(x_t, i_t)$ были также линейны, то не учитывая условия целочисленности (6.3), для поиска оптимального решения можно было бы решить динамическую задачу линейного программирования (см. п.3.1.3). Однако при решении большинства подобных задач оперативного планирования функции затрат нелинейны. Это происходит потому, что для выпуска партии изделий обычно требуются дорогостоящие подготовительные операции, из-за которых затраты на производство первой единицы партии превышают затраты на производство любой следующей единицы (рис. 70).

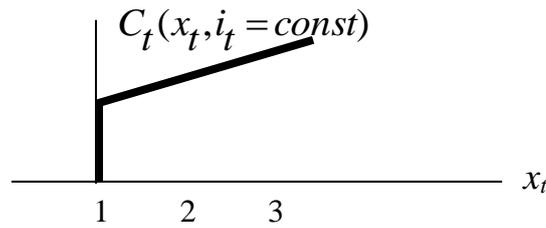


Рис. 70. Зависимость функции затрат от объема выпуска

Задача (6.2) - (6.4) есть нелинейная задача, на решение которой наложено условие целочисленности. Чтобы решить ее, запишем эту задачу в терминах динамического программирования. Для этого вместо того, чтобы рассматривать модель (6.2) - (6.4) целиком, мы будем рассматривать поведение исследуемой системы на каждом из временных отрезков по отдельности. С учетом обратного порядка рассмотрения этапов планирования введем новые обозначения для параметров задачи:

d_n – спрос на продукцию на временном отрезке, отстоящем от конца планового периода на n отрезков (включая рассматриваемый);

$c_n(x, i)$ – затраты на отрезке n (с конца), обусловленные выпуском на данном отрезке x единиц продукции и содержанием запасов в объеме i на конец отрезка n . То есть $d_1 \equiv D_N, \dots, d_N \equiv D_1$, $c_1(x, i) \equiv C_N(x_N, i_N)$. Возможность опустить индексы у переменных x и i обусловлена тем, что для различных этапов переменные x и i будут рассматриваться раздельно.

Для задачи управления запасами состояние S системы в начале любого отрезка определяется имеющимся объемом запасов на начало этого отрезка. Из принципа оптимальности следует, что при принятии решения об объеме выпуска на каком-то отрезке не нужно знать, каким образом достигнут начальный уровень запасов для этого отрезка. С учетом этого введем обозначения:

$f_n(S)$ – минимальные суммарные затраты на n отрезках с конца при условии, что объем запасов на начало отрезка n (с конца) был S .

$x_n(S)$ – оптимальный объем выпуска на отрезке n , то есть такой выпуск, который обеспечивает достижение $f_n(S)$.

Рассмотрим процесс нахождения $f_n(S)$ и $x_n(S)$. Допустим, что мы находимся в начале последнего временного отрезка (рис.71). Так как запасов на конец отрезка 1 нет, то: $f_1(S) = c_1(d_1 - S, 0)$, $S = 0, 1, \dots, d_1$.

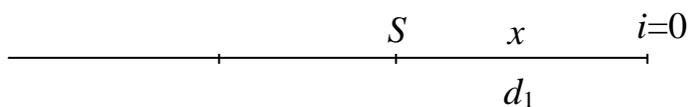


Рис. 71. Наглядное представление последнего временного отрезка

Перейдем теперь к $n=2$ (рис. 72). Если начальный уровень запасов S , а объем выпуска – x , то общие затраты для двух последних отрезков равны:

$$c_2(x, S + x - d_2) + f_1(S + x - d_2), \quad S = 0, 1, \dots, d_1 + d_2,$$

так как считаем, что стратегия на отрезке 1 была оптимальной.

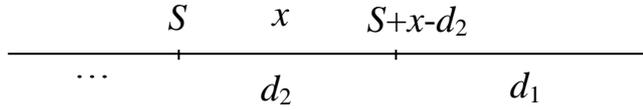


Рис. 72. Наглядное представление двух последних временных отрезков

При заданном S целочисленное значение x должно удовлетворять условию: $d_2 - S \leq x \leq d_1 + d_2 - S$. Оптимальное значение x_2 соответствует $f_2(S)$:

$$f_2(S) = \min_x [c_2(x, S + x - d_2) + f_1(S + x - d_2)],$$

$$S = 0, 1, \dots, d_1 + d_2, \quad d_2 - S \leq x \leq d_1 + d_2 - S.$$

После того, как найдено $f_2(S)$ можно вычислить $f_3(S)$ и т.д. В конце концов можно вычислить $f_N(S_N)$, где S_N – уровень запасов на начало планового периода. Таким образом, оптимальный объем выпуска на любом шаге n – x_n удовлетворяет рекуррентному соотношению динамического программирования:

$$f_n(S) = \min_x [c_n(x, S + x - d_n) + f_{n-1}(S + x - d_n)],$$

$$n = \overline{1, N}, \quad S = 0, 1, \dots, d_1 + \dots + d_n, \quad (6.5)$$

$$d_n - S \leq x \leq d_1 + d_2 + \dots + d_n - S.$$

Пример. Допустим, что спрос на продукцию постоянен во времени: $D_t=3$ ($t=1,2,3,4$) (рис.73). Затраты на отрезке t равны: $C_t(x_t, i_t)=C(x_t)+hi_t$, где $C(0)=0$, $C(1)=15$, $C(2)=17$, $C(3)=19$, $C(4)=21$, $C(5)=23$, $h=1$. То есть затраты на производство можно рассматривать как сумму постоянных затрат на операции по переналадке (13 единиц) и пропорциональных затрат - 2 единицы на каждую единицу продукции. Затраты на содержание запасов линейны.

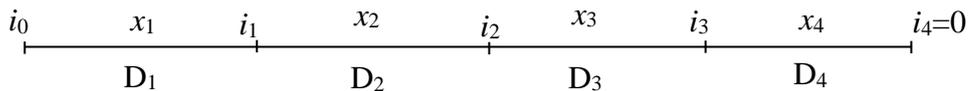


Рис. 73. Наглядное представление планового периода

Известно, что производственные мощности предприятия и складские площади ограничены. При этом выпуск в течение одного отрезка не может

превысить 5 единиц, а уровень запасов на конец отрезка – 4 единицы. То есть на переменные наложены ограничения:

$$\begin{aligned}x_t &= 0, 1, \dots, 5 \quad (t = \overline{1, 4}), \\i_t &= 0, 1, \dots, 4 \quad (t = \overline{0, 1, 2, 3}), \\i_4 &= 0.\end{aligned}$$

Сформулируем модель динамического программирования, то есть запишем соответствующее рекуррентное соотношение:

$$f_n(S) = \min_x [c_n(x, S + x - 3) + f_{n-1}(S + x - 3)] = \min_x [C(x) + 1 \cdot (S + x - 3) + f_{n-1}(S + x - 3)],$$

$$n = \overline{1, 4}, \quad S = 0, 1, 2, 3, 4, \quad 3 - S \leq x \leq \min(5, 7 - S).$$

x не может быть больше, чем $7 - S$, так как в этом случае уровень запасов на конец отрезка превысит 4, что недопустимо.

Допустим, что $n = 1$. Тогда $f_1(S) = \min_x [C(x)] = \min_x [C(3 - S)] = C(3 - S)$.

Соответствующие расчеты приведены на рис. 74.

n=1		
s	$x_1(S)$	$f_1(S)$
0	3	19
1	2	17
2	1	15
3	0	0

Рис. 74. Оптимальный объем выпуска $x_1(S)$ на последнем этапе

Результаты расчетов для последующих шагов $n=2, 3, 4$ приведены соответственно на рис. 75, 76, 77.

n=2								
$s \backslash x$	0	1	2	3	4	5	$x_2(s)$	$f_2(s)$
0	×	×	×	19+0+19	21+1+17	23+2+15	3	38
1	×	×	17+0+19	19+1+17	21+2+15	23+3+0	5	26
2	×	15+0+19	17+1+17	19+2+15	21+3+0	×	4	24
3	0+0+19	15+1+17	17+2+15	19+3+0	×	×	0	19
4	0+1+17	15+2+15	17+3+0	×	×	×	0	18

Рис. 75. Оптимальный объем выпуска $x_2(S)$ на предпоследнем этапе

		$n=3$							
$s \backslash x$	0	1	2	3	4	5	$x_3(s)$	$f_3(s)$	
0	×	×	×	19+38	22+26	25+24	4	48	
1	×	×	17+38	20+26	23+24	26+19	5	45	
2	×	15+38	18+26	21+24	24+19	27+18	4	43	
3	0+38	16+26	19+24	22+19	25+18	×	0	38	
4	1+26	17+24	20+19	23+18	×	×	0	27	

Рис. 76. Оптимальный объем выпуска $x_3(S)$ на 3-м этапе от конца

		$n=4$							
$s \backslash x$	0	1	2	3	4	5	$x_4(s)$	$f_4(s)$	
0	×	×	×	19+48	22+45	25+43	3,4	67	
1	×	×	17+48	20+45	23+43	26+38	5	64	
2	×	15+48	18+45	21+43	24+38	27+27	5	54	
3	0+48	16+45	19+43	22+38	25+27	×	0	48	
4	1+45	17+43	20+38	23+27	×	×	0	46	

Рис. 77. Оптимальный объем выпуска $x_4(S)$ на 4-м этапе от конца

После того, как вычислены все рекуррентные соотношения, нетрудно найти оптимальный план выпуска при любом начальном уровне запасов двигаясь от $n=4$ к $n=1$. Допустим, что на начало планового периода имеется 2 единицы запасов. Из рис.77 ($n=4$) находим, что $x_4(2)=5$, $f_4(2)=54$. Так как на начало отрезка $n=3$ имеем запасов $2+5-3=4$ единицы, то из рис.76 находим: $x_3(4)=0$. Аналогично, при $n=2$ $x_2(1)=5$, а при $n=1$ $x_1(3)=0$.

Следует отметить, что оптимальное решение задачи управления запасами, а также других задач, рассматриваемых в данном разделе, может быть найдено не только методом динамического программирования, но и методом нахождения кратчайшего пути на ациклической сети. Эта сеть строится следующим образом. Для каждого временного отрезка вводится столбец узлов (рис. 78).

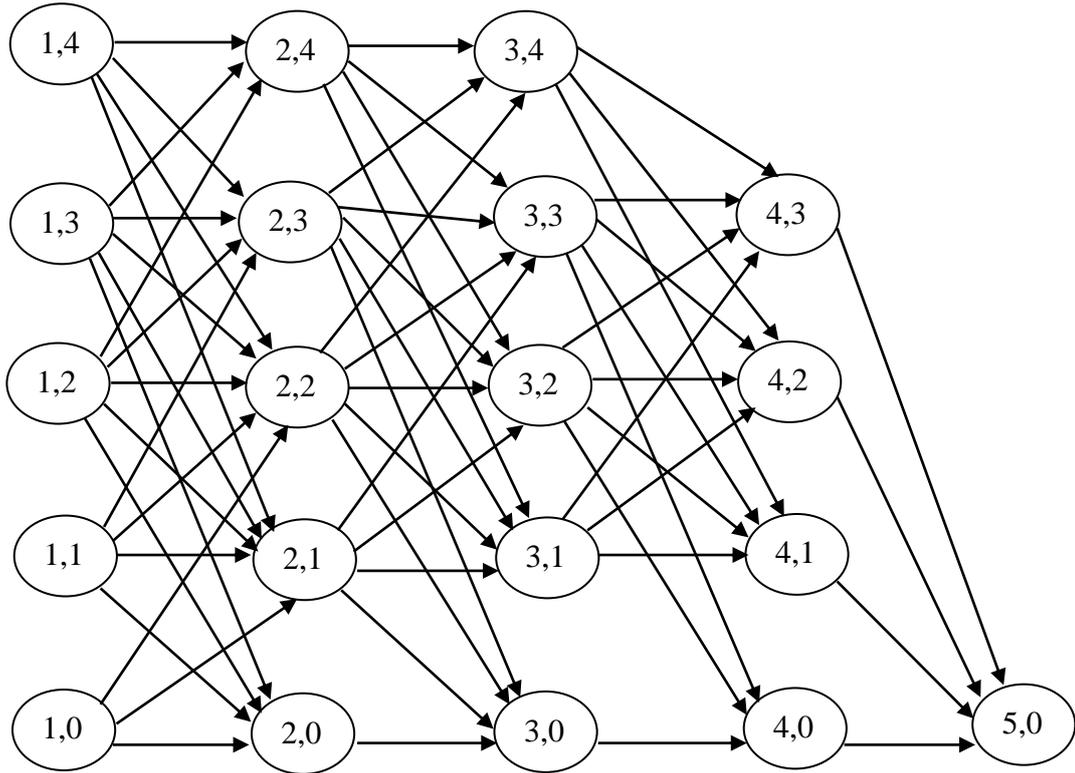


Рис. 78. Пример ациклической сети для задачи управления запасами

В обозначении узла (t, S) t есть номер отрезка (от начала), S – уровень запасов на начало этого отрезка. Каждая дуга, выходящая из узла (t, S) соответствует конкретному решению по выбору x_t . В зависимости от начального объема запасов необходимо найти кратчайший путь в вершину $(5,0)$ или из вершины $(1,4)$, или из вершины $(1,3)$ и т.д.

Преимущество метода динамического программирования по сравнению с поиском кратчайшего пути на ациклической сети состоит в меньшем объеме требуемой машинной памяти. Нетрудно заметить, что каждая незапрещенная клетка любой из приведенных выше таблиц однозначно соответствует своей дуге сети. Но в то время, как при применении сетевого метода необходимо хранить c_{ij} для всех дуг, в методе динамического программирования после завершения очередного шага достаточно оставить в машинной памяти только 2 последних столбца соответствующей таблицы. После завершения алгоритма в памяти хранится только одна таблица (рис.79).

S	$n=1$		$n=2$		$n=3$		$n=4$	
	$x_1(S)$	$f_1(S)$	$x_2(S)$	$f_2(S)$	$x_3(S)$	$f_3(S)$	$x_4(S)$	$f_4(S)$
0	3	19	3	38	4	48	3,4	67
1	2	17	5	26	5	45	5	64
2	1	15	4	24	4	43	5	54
3	0	0	0	19	0	38	0	48
4	×	×	0	18	0	27	0	46

Рис.79. Итоговая таблица примера управления запасами

Другим преимуществом метода динамического программирования является его пригодность для проведения анализа на чувствительность.

6.3 Модель распределения ресурса

6.3.1 Частная модель

Метод динамического программирования может быть применен не только для решения динамических задач, в которых значения управляемых переменных меняются со временем. Данный многошаговый метод применим также для решения статических задач, когда речь идет о распределении какого-то ресурса, например, денежных средств, между несколькими объектами. В этом случае на каждом шаге динамического программирования решается задача о выделении ресурса какому-то одному из объектов.

Немного расширим постановку задачи распределения капиталовложений, решавшуюся в п. 4.3.3 путем построения ациклической сети. Пусть имеется B единиц какого-то ресурса, который требуется распределить среди N объектов таким образом, чтобы получить от этих объектов максимальную суммарную прибыль. Известно, что если выделить объекту i x_i единиц ресурса, то он даст прибыль в объеме $C_i(x_i)$, где C_i – некоторая нелинейная функция.

Математическая модель имеет вид:

$$\sum_{i=1}^N C_i(x_i) \Rightarrow \max, \quad (6.6)$$

$$\sum_{i=1}^N x_i \leq B, \quad (6.7)$$

$$x_i = 0, 1, 2, \dots \quad (i = \overline{1, N}). \quad (6.8)$$

Запишем задачу (6.6) - (6.8) в терминах динамического программирования. Для этого вместо того, чтобы рассматривать модель (6.6) - (6.8) целиком, мы будем рассматривать распределение ресурса для каждого из объектов по отдельности. В отличие от операций, этапы которых протекают во времени, порядок рассмотрения объектов капиталовложений не важен. Тем не менее для единообразия изложения мы будем рассматривать в первую очередь объект N , затем объект $N-1$ и т.д. Для этого пронумеруем объекты в обратном порядке. С учетом новой нумерации объектов введем новые обозначения для параметров задачи:

$c_n(x)$ – прибыль от вложения в объект n x единиц ресурса. То есть $c_1(x) \equiv C_N(x_N)$. Возможность опустить индекс у переменной x обусловлена тем, что для различных объектов вложений переменные x будут рассматриваться отдельно.

Для задачи распределения ресурса состояние S системы в начале любого этапа определяется имеющимся объемом ресурса на начало этого этапа. Из принципа оптимальности следует, что при принятии решения об объеме вложений в какой-то объект не нужно знать, каким образом достигнут начальный уровень ресурса для этого этапа. С учетом этого введем обозначения:

$f_n(S)$ – прибыль при оптимальном распределении S единиц ресурса среди n объектов;

$x_n(S)$ – оптимальный объем средств, вложенный в объект n , если на все n объектов выделены S единиц ресурса, то есть такой объем, который обеспечивает достижение $f_n(S)$.

Рекуррентное соотношение динамического программирования, соответствующее задаче (6.6) - (6.8) имеет следующий вид:

$$f_n(S) = \max_x [c_n(x) + f_{n-1}(S - x)], \quad (6.9)$$

$$n = \overline{1, N}, \quad S = 0, 1, \dots, B, \quad x = 0, 1, \dots, S.$$

Основная сущность соотношения (6.9) заключается в том, что для нахождения оптимального распределения ресурса среди n объектов достаточно найти лишь оптимальное значение ресурса, выделяемого объекту n , учитывая найденное ранее оптимальное распределение ресурса среди $(n-1)$ объектов.

6.3.2. Модель общего вида

Приведенный выше пример оптимального распределения является частным случаем общей задачи распределения. Частный характер приведенной задачи обусловлен тем, что единственное ограничение (6.7) является линейным, причем с коэффициентами при переменных, равными +1. Точно также реализуется метод динамического программирования и в том случае, когда единственное ограничение нелинейно. Соответствующая модель имеет вид:

$$\sum_{i=1}^N C_i(x_i) \Rightarrow \max, \quad (6.10)$$

$$\sum_{i=1}^N H_i(x_i) \leq B, \quad (6.11)$$

$$x_i = 0, 1, 2, \dots \quad (i = \overline{1, N}). \quad (6.12)$$

В модели (6.10) - (6.12) предполагается, что каждая функция $H_i(x_i)$ есть некоторая нелинейная функция, удовлетворяющая условию: $H_i(0) = 0$.

Рекуррентное соотношение динамического программирования, соответствующее задаче (6.10) - (6.12), имеет вид:

$$f_n(S) = \max_x [c_n(x) + f_{n-1}(S - H_n(x))], \quad (6.13)$$

$$n = \overline{1, N}, \quad S = 0, 1, \dots, B, \quad H_n(x) \leq S.$$

Для нахождения оптимального решения необходимо определить согласно (6.13) значение каждой функции $f_n(S)$ при $S=0, 1, \dots, B$, начиная с $n=1$ и кончая $n=N$. Далее из обратного просмотра, начиная от $n=N$ и кончая $n=1$ находят оптимальные значения x_n , соответствующие конкретному значению B .

Пример. Пусть рассматривается возможность строительства трех объектов 1,2,3. В каждый из объектов может быть вложено 0,1,2 или 3 млн. рублей. В зависимости от объема средств, вложенных в объект, его прибыльность будет различной:

$$C_1(0)=0, \quad C_1(1)=9, \quad C_1(2)=13, \quad C_1(3)=15.$$

$$C_2(0)=0, \quad C_2(1)=5, \quad C_2(2)=9, \quad C_2(3)=11,$$

$$C_3(0)=0, \quad C_3(1)=3, \quad C_3(2)=5, \quad C_3(3)=8,$$

Чем больший объем средств вложен в объект, тем больше количество человек будет работать на нем после завершения строительства. Пусть каждый миллион рублей, вложенных в объект 1,2,3, приводит, соответственно, к появлению 4-х, 3-х и 2-х десятков новых рабочих мест. Допустим, что общее число новых рабочих мест не должно превышать 8 десятков. Требуется найти оптимальное распределение средств между объектами.

Модель, описывающая данную задачу, имеет вид:

$$C_1(x_1) + C_2(x_2) + C_3(x_3) \Rightarrow \max,$$

$$4x_1 + 3x_2 + 2x_3 \leq 8,$$

$$x_i = 0, 1, 2, 3 \quad (i=1, 2, 3).$$

Запишем рекуррентное соотношение динамического программирования:

$$f_n(S) = \max_x [c_n(x) + f_{n-1}(S - H_n(x))],$$

$$n = 1, 2, 3, \quad S = 0, 1, \dots, 8, \quad H_n(x) \leq S,$$

$$\text{где } H_n(x) = \begin{cases} 2x, & n = 1, \\ 3x, & n = 2, \\ 4x, & n = 3. \end{cases}$$

Допустим, что мы рассматриваем вложение средств только в объект 1:

$$f_1(S) = \max_x [c_1(x)], \quad 2x \leq S, \quad S = 0, 1, \dots, 8.$$

Результаты расчетов на данном шаге представлены на рис.80. Результаты расчетов при $n=2$ и $n=3$ представлены, соответственно, на рис.81 и 82.

	$n=1$								
S	0	1	2	3	4	5	6	7	8
$x_1(S)$	0	0	1	1	2	2	3	3	3
$f_1(S)$	0	0	3	3	5	5	8	8	8

Рис. 80. Расчет оптимального вложения средств в объект 1 (с конца)

$S \backslash x$	0	1	2	3	$x_2(S)$	$f_2(S)$
0	0	×	×	×	0	0
1	0	×	×	×	0	0
2	0+3	×	×	×	0	3
3	0+3	5+0	×	×	1	5
4	0+5	5+0	×	×	0, 1	5
5	0+5	5+3	×	×	1	8
6	0+8	5+3	9+0	×	2	9
7	0+8	5+5	9+0	×	1	10
8	0+8	5+5	9+3	×	2	12

Рис. 81. Расчет оптимального вложения средств в объект 2 (с конца)

$S \backslash x$	0	1	2	3	$x_3(S)$	$f_3(S)$
0	0	×	×	×	0	0
1	0	×	×	×	0	0
2	0+3	×	×	×	0	3
3	0+5	×	×	×	0	5
4	0+5	9+0	×	×	1	9
5	0+8	9+0	×	×	1	9
6	0+9	9+3	×	×	1	12
7	0+10	9+5	×	×	1	14
8	0+12	9+5	13+0	×	1	14

Рис. 82. Расчет оптимального вложения средств в объект 3 (с конца)

Из таблиц на рис. 80-82 двигаясь в обратном направлении от $n=3$ к $n=1$ нетрудно найти оптимальное вложение средств в каждый объект, учитывая размер константы в правой части единственного ограничения. Если она равна 8, то из таблицы $n=3$ при $S=8$ следует, что $x_3=1$. Из таблицы $n=2$ при $S=4$ следует, что $x_2=0$ или 1. Если $x_2=0$, то из таблицы $n=1$ при $S=4$ следует, что $x_1=2$, а если $x_2=1$, то $x_1=0$.

Следует отметить, что данная задача может быть решена не только методом динамического программирования, но и методом нахождения кратчайшего пути на ациклической сети. Соответствующая сеть строится так же, как и в п.4.3.3.

Описанный подход нахождения оптимального решения нелинейной модели с одним ограничением может быть распространен на модели с большим числом ограничений. В этом случае состояние системы в начале каждого шага описывается не одной, а несколькими (по числу ограничений модели) переменными состояниями. Сама же сущность метода остается той же.

6.4 Анализ на чувствительность

В качестве примера рассмотрим анализ на чувствительность решенной в п.6.2 численной задачи управления запасами. Для выполнения данного анализа будет использоваться итоговая таблица, находящаяся в машинной памяти после завершения алгоритма динамического программирования (см. рис.79).

6.4.1 Длительность планового периода

Допустим, что для ЛПР важно знать, как будут изменяться оптимальные месячные объемы выпуска при изменении числа месяцев N в плановом периоде, который, для определенности, начинается в январе. В частности, для ЛПР особый интерес представляет январский выпуск продукции.

Если известно, что уровень запасов на начало января 0, то результаты анализа можно представить в виде таблицы (рис. 83). Рассмотрим, например, как получена строка этой таблицы при $N=3$.

Плановый период в этом случае включает январь, февраль, март. Из таблицы на рис.83 при $S=0$ и $n=3$ находим, что оптимальное производство в январе составляет 4 единицы, а суммарные оптимальные затраты на все три месяца – 48 единиц. Так как в январе производим 4, то запасы на начало февраля равны 1. Из рис.79 при $S=1$ в столбце $n=2$ находим, что оптимальный объем производства в феврале 5 единиц. Так как запасы на начало марта – 3, то из рис.79 при $S=3$ в столбце при $n=1$ находим, что объем производства в марте равен 0.

Проведенный анализ на чувствительность позволяет сделать вывод о том, что оптимальный объем производства в январе при увеличении длительности планового периода с 1 до 4-х месяцев изменяется слабо, хотя и имеет некоторую тенденцию к увеличению. Среднемесячные затраты с ростом N сначала имеют тенденцию к уменьшению, а затем вновь возрастают.

N	Январь	Февраль	Март	Апрель	$f_N(0)$	$f_N(0)/N$
1	3	×	×	×	19	19
2	3	3	×	×	38	19
3	4	5	0	×	48	16
4	3 4	4 5	5 0	0 3	67	$16\frac{3}{4}$

Рис. 83. Месячные объемы выпуска и среднемесячные затраты при различных N ($S=0$)

6.4.2 Исходный уровень запасов

Изучим зависимость оптимальной программы выпуска от уровня запасов на начало планового периода. В частности, нас интересует оптимальный выпуск в январе. Результаты анализа на чувствительность представлены в таблице на рис. 84.

N	Выпуск в январе			Выгода от единичного увеличения запаса	
	$S_0=0$	$S_0=1$	$S_0=2$	$S_0=0 \rightarrow S_0=1$	$S_0=1 \rightarrow S_0=2$
1	3	2	1	2	2
2	3	5	4	12	2
3	4	5	4	3	2
4	3, 4	5	5	3	10

Рис. 84. Зависимость январского выпуска от N и S_0

Если плановый период включает всего лишь один месяц, каждое увеличение исходного запаса на 1 единицу приводит к снижению январского выпуска. Однако при $N=2,3,4$ увеличение исходного запаса от 0 до 1 приводит не к сокращению, а к росту выпуска в январе. Увеличение же исходного запаса с 1 до 2 приводит к различным результатам в зависимости от N . При этом выпуск продукции может сократиться или остаться прежним.

В двух последних столбцах отражено снижение общей суммы затрат при единичном приращении исходного запаса. Пусть, например, длительность

планового периода $N=2$. Из таблицы на рис.79 видно, что при нулевом исходном уровне запасов общая сумма затрат составляет 38. Наличие 1 единицы исходного запаса позволяет снизить затраты до 26, а 2-х единиц до 24. Таким образом, выгода от первого единичного приращения исходного запаса 12, а второго – только 2.

6.5. Заключение

Метод динамического программирования является одним из самых распространенных методов исследования операций. Произведем его сравнение с другими методами.

По сравнению с методом перебора всех вариантов данный метод обеспечивает огромную экономию машинного времени. В качестве примера рассмотрим приведенную в п. 6.1 задачу нахождения кратчайшего пути. В данной задаче 14 возможных путей. Чтобы найти длину каждого из них необходимо произвести 4 операции сложения, то есть всего $14 \times 4 = 56$ операций суммирования. В приведенных же таблицах динамического программирования выполнено всего 16 операций сложения. С увеличением размерности задачи данное преимущество может быть огромным.

В каждой из приведенных выше задач динамического программирования состояние исследуемой системы описывается в начале очередного шага с помощью лишь одной переменной состояния S . В принципе данный метод позволяет использовать большее число переменных состояния, то есть S может быть вектором. Сущность используемого метода при этом остается той же за исключением того, что количество строк в таблицах резко возрастает. Поэтому на практике метод динамического программирования применяется лишь при небольшом числе переменных состояния, что эквивалентно небольшому числу ограничений в исходной постановке задачи.

Поэтому в отличие от линейного программирования, которое широко применяется для решения задач стратегического планирования, имеющих многие десятки ограничений, метод динамического программирования широко используется при решении задач оперативного планирования, которые часто бывают нелинейными и имеют небольшое число ограничений.

Как было показано на примерах, любая задача, решаемая методом динамического программирования, в принципе может быть решена путем построения соответствующей сети и нахождения на ней кратчайшего пути. Преимуществом метода динамического программирования является меньший объем требуемой памяти.

Достоинством динамического программирования по сравнению с целочисленным программированием является возможность решать нелинейные задачи. Недостаток – меньшее число ограничений.

ЛИТЕРАТУРА

- 1 Вагнер Г. Основы исследования операций. Том 1,2,3. – М.: Мир, 1973.
- 2 Аронович А.Б. Сборник задач по исследованию операций. – М.: Изд-во Московского ун-та, 1997.
- 3 Хэмди А. Введение в исследование операций. – М.: «Вильямс», 2001.
- 4 Вентцель Е.С. Исследование операций. – М.: Высшая школа, 2001.
- 5 Конюховский П. Математические методы исследования операций в экономике. – СПб.: «Питер», 2002.