

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА**

Кафедра «ПРИКЛАДНАЯ МАТЕМАТИКА»

С. И. ЖОГАЛЬ, И. В. МАКСИМЕЙ

ЗАДАЧИ И МОДЕЛИ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ

Часть 1

**Аналитические модели
исследования операций**

Гомель 1999

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА**

Кафедра «ПРИКЛАДНАЯ МАТЕМАТИКА»

С. И. ЖОГАЛЬ, И. В. МАКСИМЕЙ

ЗАДАЧИ И МОДЕЛИ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ

Часть 1

Аналитические модели исследования операций

*Допущено Министерством образования Республики Беларусь
в качестве учебного пособия
для студентов инженерно-технических специальностей
высших учебных заведений*

Под общей редакцией профессора **И. В. Максimea**

Гомель 1999

УДК 519.85 (075.8)
Ж 783

С. И. Жогаль, И. В. Максимей

Ж 783 Задачи и модели исследования операций. Ч. 1. Аналитические модели исследования операций: Уч. пособие. – Гомель: БелГУТ, 1999. – 110 с.

ISBN 985-6550-10-6

Содержатся технологии формализации и методы решения производственных задач с помощью аналитических моделей: линейного программирования, сетевых и графовых, календарного планирования, динамического программирования, теории игр. По каждому классу аналитических моделей приводятся практические примеры производственного характера.

Предназначено для студентов и аспирантов инженерно-технических специальностей высших учебных заведений.

Рецензенты: **кафедра «Прикладная математика»** Национальной политехнической академии Украины «Киевский политехнический институт»;
докт. техн. наук профессор Белорусского государственного университета **О. М. Тихоненко**;
проректор БелГУТа докт. техн. наук профессор **В. Я. Негрей**.

Учебное издание

Светлана Ивановна Жогаль
Иван Васильевич Максимей

ЗАДАЧИ И МОДЕЛИ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ Часть 1. Аналитические модели исследования операций

Учебное пособие

Редактор Н. А. Дашкевич
Технический редактор Т. А. Пугач
Корректор И. И. Эвентов

Подписано в печать 25.03.99.

Формат бумаги $60 \times 84 \frac{1}{16}$. Бумага офсетная № 1. Гарнитура Таймс.

Печать офсетная. Усл. печ. л. 6,51 Уч.-изд. л. 7,72. Тираж 200 экз.
Зак. № . Изд. № 3082

Редакционно-издательский отдел БелГУТа,
246653, г. Гомель, ул. Кирова, 34. Лицензия ЛВ № 57 от 22.10.97.

Ротапринт типографии БелГУТа,
246022, г. Гомель, ул. Кирова, 34. Лицензия ЛП № 75 от 12.03.93.

ISBN 985-6550-10-6

© С. И. Жогаль, И. В. Максимей, 1999.

ВВЕДЕНИЕ

Исследование операций (ИСО) является сравнительно молодой наукой, которая появилась во время второй мировой войны и использовалась для подготовки оптимальных решений командиров. Были созданы специальные группы математиков и инженеров, которые рассматривали вопросы применения оружия и распределения сил и средств по военным объектам. Затем ИСО расширило область применения на промышленность, сельское хозяйство, бытовое обслуживание, охрану природы и транспорт. Особенно актуально применение методов формализации ИСО на транспорте и в строительстве.

При всём многообразии содержания конкретных работ по ИСО каждое операционное исследование проходит последовательно следующие основные этапы: постановка задачи, построение математической модели, нахождение решения, проверка и корректировка модели, реализация найденного решения на практике. Постановка задачи – весьма ответственный этап при анализе транспортных потоков и оптимизации технологических процессов строительства объектов и транспортных коммуникаций. Цель данного пособия – обучение студентов вопросам формализации реальных процессов. Первоначально задачу формулируют с точки зрения заказчика. Поэтому состав примеров и структура курса ИСО ориентированы на первые два этапа операционного исследования. Любая постановка задачи обычно не бывает окончательной. Во время анализа исследуемой системы задача постепенно уточняется. Весьма существенно провести исследование объекта, изучить множество факторов, влияющих на результаты анализа. Исходя из этих проблем в пособии по каждому разделу предлагаются соответствующие технологические приёмы перехода от содержательной постановки задачи к её формальному изложению. Как правило, это самый трудный вопрос во всём цикле исследований. Состав примеров в пособии подобран исходя из необходимости освоения ряда технологических подходов к построению моделей объекта исследования.

Второй важный этап ИСО – построение математической модели (ММ). Получив достаточно строгую и логически непротиворечивую содержательную постановку задачи, переходят к её формализации. Каждый класс ММ в

ИСО ориентирован на использование одного или нескольких способов формализации, и для него в ИСО разработано большое многообразие методов их решения. В пособии предложены наиболее известные методы, которые широко распространены и обладают необходимыми технологическими возможностями для решения задач на транспорте и в строительстве.

Исходя из сформулированных целей, первый том пособия содержит технологии формализации и методы решения производственных задач с помощью следующих формализмов: модели линейного программирования, сетевые и графовые модели технических систем, модели динамического программирования, игровые модели ИСО. Как видим, в пособии представлен основной состав аналитических детерминированных ММ. Кроме того, в пособии излагается известный кибернетический подход к исследованию технических систем. Перечисляются также другие способы формализации технических систем и приводятся основные отличительные особенности ИСО.

В данном пособии кандидату технических наук С. И. Жогаль принадлежат все практические примеры, остальной текст и научное редактирование выполнены доктором технических наук профессором И. В. Максимеем.

Авторы выражают благодарность рецензентам пособия за ценные советы и замечания и надеются, что пособие поможет студентам освоить основы ИСО и сэкономит их время на подготовку и приобретение навыков системного анализа технических систем.

1. ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

1.1. Кибернетический подход к исследованию технических систем

Обычно математические модели (ММ) представляют собой формализованную запись процессов, происходящих в объектах исследования, и служат как для исследования свойств этих объектов, так и для предсказания их поведения в различных ситуациях. Если исходить из *соотношений*, которые выражают зависимости между состояниями и параметрами объектов исследования (ОИ), различают модели *детерминированные* и *вероятностные* (стохастические). В детерминированных ММ при совместном рассмотрении этих соотношений отклики системы (\vec{Y}), состояния системы (\vec{S}) в каждый данный момент времени t определяются через её параметры (\vec{p}), входную информацию (\vec{X}) и начальные условия (\vec{U}). Составляются однозначно определённые функциональные зависимости откликов и состояний в явном виде:

$$\begin{aligned}\vec{Y} &= \Phi_1(\vec{X}, \vec{U}, \vec{p}); \\ \vec{S}_i &= \Phi_2(\vec{S}_{i-1}, \vec{X}, \vec{U}, \vec{p}).\end{aligned}\quad (1.1)$$

В вероятностных ММ с помощью упомянутых соотношений (1.1) можно определить лишь распределения вероятностей для компонент вектора состояний $F_{1i}(S_i)$ и вектора откликов $F_{2j}(Y_j)$, если заданы распределения вероятностей для начальных условий $F_{3i}(U_i)$, её параметров $F_{4i}(\pi_i)$ и входной информации $F_{5j}(X_j)$.

Если исходить из способа дальнейшего использования ММ для изучения ОИ, то модели можно разделить на аналитические, численные и имитационные. Для *аналитических* ММ процессы функционирования технических систем (ТС) записываются в виде некоторых функциональных соотношений (алгебраических, интегрально-дифференциальных). При этом всегда стремятся получить в общем виде явные зависимости для искомых величин. Поэтому предсказательные возможности аналитических ММ очень велики. Если математические зависимости сложны, то зачастую используются ЭВМ, позволяющие быстро вычислить значения компонент векторов \vec{S} и \vec{Y} для любых заданных значений векторов \vec{X} , \vec{U} , \vec{p} или их распределений (для вероятностных ММ).

Для *численных* ММ, когда в общем виде трудно найти явные зависимо-

сти для искомых величин в уравнениях (1.1), стараются получить числовые результаты при конкретных начальных расчётных данных компонент векторов \vec{X} , \vec{U} и \vec{p} . В данном случае широко используется алгоритмическое описание процесса функционирования ТС и её компонентов. Результат получается путём использования численных методов решения зависимостей Φ_1 и Φ_2 . Причём в явном виде эти зависимости удаётся получить на практике весьма редко. Поэтому исследователи вначале стремятся получить аналитическое решение задачи. При этом они идут умышленно на упрощение реальной ситуации, чтобы иметь возможность изучать некоторые общие свойства системы. В отдельных случаях приближённое решение задачи о поведении ТС исследователей удовлетворяет. Поэтому для них достаточно результатов, полученных с помощью качественных методов на аналитической ММ.

Для получения аналитического или численного решения различных типов функциональных уравнений (1.1) создан мощный математический аппарат: алгебра; функциональный анализ; теория и методы решения дифференциальных, интегральных и разностных уравнений; теория вероятностных процессов; численные методы; методы оптимизации и т. д.

Имитационные ММ применяются тогда, когда ТС особенно сложна или когда необходим высокий уровень детализации представления процессов, протекающих в ней. Для таких ТС ради получения аналитической ММ исследователь вынужден накладывать жёсткие ограничения на модель и прибегать к упрощениям. При этом приходится пренебрегать некоторыми особенностями ТС, что приводит к тому, что ММ перестаёт быть средством изучения СС. В имитационных моделях (ИМ) моделируемый алгоритм поведения ТС приближённо воспроизводит сам процесс-оригинал в смысле его функционирования во времени. При этом имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры и порядка протекания во времени. Таким образом, реализуется на ЭВМ специальный алгоритм, который воспроизводит формализованный процесс поведения ТС. Этот алгоритм по исходным данным (\vec{U} , \vec{p} , \vec{X}) позволяет получить информацию об изменении во времени t состояний \vec{S} и откликов \vec{Y} модели. В этом алгоритме можно выделить три функциональные части: моделирование элементарных подпроцессов; учёт их взаимодействия и объединение их в единый процесс; обеспечение согласованной работы отдельных подпроцессов при реализации ММ на ЭВМ. Влияние случайных факторов на течение процесса имитируется с помощью генераторов случайных чисел с заданными вероятностными характеристиками. В ходе имитации постоянно фиксируется статистика о состояниях системы \vec{S} и изменениях откликов \vec{Y} . Эта статистика либо должным образом обрабатывается в ходе имитации, либо накапливается и по окончании заданного интервала модели-

рования T_M обрабатывается статистическими методами. Как видим, идея имитации привлекательна по своей простоте, но дорога по реализации. Поэтому применяются ИМ только в тех случаях, когда другие способы моделирования неэффективны.

1.2. Способы формализации технических систем

Наиболее ранним способом формализации ТС является представление физических явлений с помощью **систем дифференциальных уравнений**. На основании известных аналогий удаётся получить явный вид функций Φ_1 и Φ_2 в соотношениях вида (1.1). Примером такой аналогии могут служить гармонические колебания. Пусть $\xi(t)$ – отклонение центра масс (ЦМ) пружинного маятника от положения равновесия в момент t , m – его масса, $\Psi_0 \xi(t)$ – сила, действующая на маятник пружины. (Ψ_0 – жёсткость пружины). Тогда дифференциальное уравнение колебания маятника имеет вид

$$m \frac{d^2 \xi(t)}{dt^2} = -\Psi_0 \xi(t).$$

Обозначив $\xi_0^2 = \frac{\Psi_0}{m}$, $\xi(t) = z$, получим общий вид уравнения гармонических колебаний

$$\frac{d^2 z}{dt^2} + \xi_0^2 z = 0. \quad (1.2)$$

Итак, мы получили аналитическую ММ, позволяющую прогнозировать поведение маятника в зависимости от параметров m и Ψ (массы и жёсткости пружины). Как видим, в ММ данного класса, в зависимости от сложности процесса, исследователь получает либо численное решение, либо аналитическое. Обычно ММ данного класса применяются для выполнения технических расчётов.

Вторым способом формализации ТС является аппарат математического программирования. В общем случае ММ математического программирования имеют вид

$$\text{найти } \max E = f(\vec{X}, \vec{Z}) \quad (1.3)$$

$$\text{при ограничениях } G_i(\vec{X}, \vec{Z}) \leq b_j; \quad i = \overline{1, m},$$

где $E = f(\vec{X}, \vec{Z})$ – целевая функция (ц. ф.) или показатель эффективности; \vec{X}, \vec{Z} – векторы входных величин и неуправляемых переменных; G_j – функции потребления j -го вида ресурсов; b_j – граничные значения j -го вида ресурсов. В зависимости от вида ц. ф. E и ограничений G_j различают сле-

дующие классы ММ математического программирования:

– если $f(\vec{X}, \vec{Z})$, $G_j(\vec{X}, \vec{Z})$ – линейные функции, то это ММ **линейного программирования**;

– если $f(\vec{X}, \vec{Z})$, $G_j(\vec{X}, \vec{Z})$ – нелинейные функции, то мы имеем дело с ММ **нелинейного программирования**;

– когда целевая функция $f(\vec{X}, \vec{Z})$ имеет специальную структуру, является аддитивной ($f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n f_j(x_j)$) или мультипликативной

($f(x_1, x_2, \dots, x_n) = n f_j(x_j)$) функцией, то это ММ **динамического программирования**;

– если ц. ф. $f(\vec{X}, \vec{Z})$ и ограничения $G_j(\vec{X}, \vec{Z})$ – функции-полиномы, то это ММ **геометрического программирования**;

– когда вектор неуправляемых переменных Z случаен, но имеет место зависимость вида $\max M_Z E = M_Z \{f(\vec{X}, \vec{Z})\}$ при вероятностных ограничениях типа

$$M_Z \{G_j(\vec{X}, \vec{Z})\} \leq b_j, \quad P \{G_j(\vec{X}, \vec{Z}) \leq b_j\} = 1 - \varepsilon_i,$$

где M_Z – математическое ожидание по z ; ε_i – уровень значимости по j -му ограничению, то это ММ **стохастического программирования**;

– если на переменные x_j наложено условие дискретности их значений, то мы имеем дело с ММ **дискретного программирования**;

– наконец, когда в ММ точный оптимум алгоритмическим путём найти невозможно из-за очень большого числа вариантов перебора, тогда мы имеем дело с ММ **эвристического программирования**.

В этих случаях исследователь отказывается от поиска оптимума и удовлетворяется лучшим вариантом из рассмотренных. При этом он пользуется специальными приёмами решения («эвристиками»), которые позволяют существенно сократить число рассматриваемых вариантов.

Третьим способом формализации ТС является метод сетевого планирования. ММ этого класса позволяют отображать объём предстоящих работ на предприятии или в строительной организации, их взаимосвязи и последовательность их выполнения. ММ сетевого планирования изображаются в виде единого графа, в котором узлами являются состояния комплекса работ, а дугами изображаются сами работы. Целью моделирования является поиск узких мест в комплексе работ и выделение напряжённых участков в сетевом графике их выполнения, а также поиск резервов ресурсов и времени выполнения отдельных работ для перераспределения по другим участкам. Второй

класс таких ММ – это, собственно, модели на графах, использующие для формализации аппарат теории графов. Целями моделирования, в частности, могут быть: поиск кратчайшего пути между вершинами неориентированного графа, построение графа наименьшей длины, определение максимального потока в сети.

Четвёртым способом формализации ТС являются **системы массового обслуживания** (СМО). В основу построения ММ СМО положено наличие: однотипных действий для массового потребителя, требующих простейшего обслуживания; очередей требований (TR) на обслуживание; дисциплин обслуживания TR . Очереди образуются из-за того, что поток TR на обслуживание неуправляем и случаен. Различают следующие типы СМО, определяющие поведение TR в тех случаях, когда обслуживаемое устройство или прибор СМО (ОУ) занят в момент прихода TR :

- СМО с ожиданием (ожидает в очереди);
- СМО с отказами (покидает систему без обслуживания);
- СМО смешанного типа (имеют место ограничения на время ожидания ($t_{ож}$) или длину очереди ($l_{оч}$), после чего TR покидает СМО необслуженным).

Основными элементами формализации ТС методами СМО являются TR и ОУ. При этом на ОУ имитируется задержка TR длительностью, равной времени их обслуживания ($t_{обс}$), которое в общем случае может быть случайной величиной. В последнем случае говорят, что ТС отображается стохастической СМО. Перемещение TR по СМО осуществляется по определённым правилам (дисциплинам обслуживания). Для оценки эффективности ТС, моделируемой в виде СМО, задаются различные показатели эффективности вариантов организации ТС. Весь процесс моделирования сводится к поиску узких мест в системе. В качестве критериев эффективности, с точки зрения организации ТС, ищутся места, где коэффициент использования i -го ОУ обслуживанием TR (η_i) максимален и при этом $l_{очi}$ и $t_{ожi}$ также максимальны. А с точки зрения пользователей ТС определяется время жизни TR в системе. Тот вариант ТС является лучшим, у которого среднее время жизни транзактов j -го типа ($t_{жj}$) будет минимальным. Основными варьируемыми величинами являются либо параметры входного потока (TR_i) на СМО, характеристики обслуживающих приборов ($t_{обси}$), организация самой ТС, её структура и взаимосвязи ОУ между собой, либо комбинации перечисленных характеристик ТС.

Пятым способом формализации ТС являются модели теории игр. Они используются тогда, когда две или несколько сторон преследуют разные цели и результат этого столкновения характеризуется выигрышем одной из сторон. На ММ теории игр вырабатываются рекомендации для действий сторон (противников в игре) в ходе конфликтной ситуации. Строится упрощённая игровая ММ (без приводящих факторов), которая от реальности

отличается тем, что ведётся по вполне определённым правилам. Стороны, участвующие в игре, называются игроками. Каждый из них придерживается одной или нескольких стратегий. Стратегией называют совокупность правил, определяющих выбор при каждом ходе данного игрока в зависимости от ситуации, сложившейся во время игры. Выигрыши ε_{ij} стратегии i игрока A при стратегии j игрока B составляют матрицу игры $\|e_{ij}\|$. Целью моделирования является поиск оптимальных стратегий поведения игроков, обеспечивающих им при многократных повторениях игры максимальные средние выигрыши.

Когда вероятностная природа переменных ТС \bar{X} , \bar{U} и \bar{Z} такова, что очень трудно сформулировать задачу в виде аналитической или численной ММ, то применяются вероятностные методы формализации. Различают три основных класса вероятностных ММ:

- марковские модели случайных процессов;
- модели на основе метода Монте-Карло;
- имитационные модели (ИМ).

Первый класс ММ использует специальный аппарат представления ТС, когда удаётся описать динамику её функционирования с помощью аппарата событий и перехода ТС из состояния в состояние. Задаются вероятности P_{ij} перехода ТС из i -го состояния в j -е, распределения времени нахождения ТС в этих состояниях. Целью моделирования является анализ вероятностей перехода в различные состояния из некоторого начального состояния и нахождение средних значений времён, требуемых для этого перехода.

Второй класс вероятностных ММ применяется в том случае, когда исследователю легче производить «розыгрыш» сложных явлений в ТС с помощью стандартной процедуры, чем описывать случайные процессы в ТС с помощью систем дифференциальных уравнений (которые зачастую ещё и трудно решаются). Результатом моделирования по методу статистических испытаний являются усреднённые характеристики случайного процесса в ТС. При этом число статистических испытаний (N) заранее рассчитывается из требуемой точности оценки параметров случайного процесса в ТС \bar{Y} при одном и том же значении переменных (\bar{X}, \bar{U}) во всех N опытах на ММ. Так же, как и в реальности, конкретная реализация случайного процесса в ТС при l -м опыте каждый раз складывается по-разному из-за случайного характера параметров ТС. В результате получают N копий случайного процесса, по которым затем находят основные характеристики поведения ТС:

- вероятности событий в ТС (\tilde{P}_{ij});
- математические ожидания и дисперсии отклика \bar{Y} .

Для третьего класса вероятностных ММ не требуется преобразование ал-

горитма поведения компонентов ТС в специальную систему уравнений искомых величин. Иногда достаточно имитировать сами явления с сохранением их логической структуры, последовательности их чередования во времени. В ходе имитации явлений фиксируется статистика поведения компонентов ТС, по которой затем можно определить все необходимые характеристики функционирования ТС. Поскольку результаты имитации являются также случайными реализациями процесса, то для нахождения объективных и устойчивых характеристик явления необходимо его многократное повторение с последующей статистической обработкой результатов имитации. По этой статистике можно вычислить значения любого функционала \bar{Y} , заданного на множестве реализаций ТС.

1.3. Отличительные особенности исследования операций

Из перечисленного обзора типов ММ, составляющих предмет ИСО, можно выделить следующие особенности ММ ИСО.

Системный подход, заставляющий исследователя находить связи в ТС, причём вначале усложняя задачу до тех пор, пока все взаимосвязи не будут учтены, а затем минимизируя их число для рассмотрения ТС как единого целого, в котором отброшены все несущественные связи между компонентами ТС.

Использование комплексных коллективов, в состав которых входят специалисты из разных областей знаний, чтобы обеспечить группе исследователей необходимый уровень квалификации.

Операционный метод, предполагающий построение ММ, в которых общий критерий функционирования системы в целом (W) приравнивается некоторым соотношениям f , связывающим множество управляемых переменных X_i , неуправляемых переменных α_j и неизвестных факторов (ξ_l):

$$W = f(X_i, \alpha_j, \xi_l). \quad (1.4)$$

При этом существует две категории задач ИСО: прямые и обратные. Прямые задачи отвечают на вопрос: что будет, если в заданных условиях (X_i, α_j, ξ_l) мы примем такое-то решение, и при этом позволяют вычислить значение критерия функционирования ТС W . Обратные задачи дают ответ, как следует выбирать элементы решения (X_i, α_j, ξ_l) для того, чтобы критерий W был максимальным.

Помощь и участие операционистов в практической реализации результатов моделирования ТС.

Независимо от типа ММ и метода исследований с их помощью, на содержательном уровне модели ИСО применяются в следующих областях.

Закупки: сколько, когда и у кого закупить; учитывается при этом вме-

стимость складов, возможность ухудшения материала со временем, изменение цены и т.д.

Производство: проектирование предприятий; выбор пунктов размещения и ассортимента продукции предприятия; распределение производственных заказов, которое даёт минимум затрат на их реализацию и максимум дохода от функционирования производства.

Сбыт: распределение, продажа и обслуживание потребителей; размещение оптовых складов, определение их вместимости и ассортимента их запасов; определение круга потребителей продукции из этих складов; нахождение издержек на организацию сбыта продукции; определение соотношений расходов на рекламу и премии за организацию сбыта продукции; определение спроса потребителей на товары.

Научно-исследовательские работы: размеры ассигнований на эти работы, выбор перспективных областей исследований, календарное планирование работ.

Упорядочение производства: определение критических последовательностей операций, дающих максимальное время реализации комплекса работ.

Кадры: определение оптимального состава между группами работников и их квалификацией; разработка методов найма рабочей силы; поиск способов стимулирования работников, повышающих их производительность труда.

Финансы и учёт: разработка методов получения экспресс-анализа финансового положения предприятия; разработка методик проведения ревизий; изучение кредитных стратегий.

Из перечисленного обзора областей применения ММ ИСО можно сделать общий вывод, что ИСО занимается не объектами изучения, отдельными ТС, а лишь методами их исследования.

Можно выделить следующие восемь классов операционных задач: управления запасами, распределения, массового обслуживания, упорядочения, выбора маршрута, замены оборудования, состязательные задачи, поиска. Некоторые ММ можно применять для решения задач разных классов, и в то же время ряд методов необходимо применять для решения задач одного и того же класса.

Управление запасами применяется в тех случаях, когда ресурсы ограничены и существует два вида издержек, связанных с хранением и перевозкой ресурсов. Руководителя предприятия в таких случаях интересует выбор такого соотношения между хранением и перевозками ресурсов, который обеспечит ему минимум затрат. Как правило, здесь используются ММ математического программирования.

В **задачах распределения** также рассматриваются вопросы, связанные с ресурсами, необходимыми для выполнения некоторого заданного вида ра-

бот. Однако руководителя здесь интересует возможность перераспределения ресурсов (где нужно добавить, каких именно ресурсов, в каком месте целесообразно отказаться от дефицитных ресурсов или же эти ресурсы можно заменить на менее дефицитные).

Задачи массового обслуживания реализуются с помощью ММ СМО и обычно сводятся к минимизации издержек на обслуживание, связанных с потерями от простоев оборудования и потерями заявок. В качестве средства реализации СМО используется аппарат теории вероятностей, теории дифференциальных и интегральных уравнений. Основное ограничение метода состоит в требовании экспоненциальности распределения $t_{об}$ на ОУ и пуассоновского характера входного потока TR .

Задачи упорядочения имеют основной целью выбор дисциплины обслуживания TR с таким расчётом, чтобы критерий функционирования системы был оптимальным. Используют специальные алгоритмы целочисленного программирования. Прирост критерия эффективности в таких системах достигается за счёт изменения порядка обслуживания TR .

Задача выбора маршрута формулируется следующим образом: нужно выбрать путь, проходящий через сеть городов, такой, чтобы в городе побывать только один раз, и этот маршрут характеризуется минимальной длиной или затратами. Как правило, в качестве ММ используются графовые модели.

Замена оборудования. Предполагается, что существует два типа элементов, из которых состоит ТС: с постоянно ухудшающимися характеристиками; внезапно выходящие из строя. Задача сводится к определению порядка и сроков замены элементов, при которых сводятся к минимуму общие эксплуатационные затраты и капиталовложения. Наиболее общим методом решения таких задач является динамическое программирование. При этом определяется сама целесообразность замены элементов ТС и частота этой операции.

Состязательные задачи. Это задачи теории игр. Используются при военных играх для обучения офицеров и дипломатов, для принятия решений при особых ситуациях. В качестве аппарата формализации используется ММ теории игр и деловые игры.

Наконец, **задачи поиска.** Имеется фиксированное количество ресурсов для проведения поисковых работ. Нужно принять решение: какой нужен выбор для обеспечения достоверности результатов поиска. Чем больше ресурсов привлекается, тем выше стоимость поиска, но тем меньше и ожидаемые потери от ошибки. Эти задачи являются типовыми для проведения ревизий или инспекций. Широко используется задача поиска в геологии при поиске природных ископаемых. Широко применяются методы поиска при организации различных опросов населения для выявления динамики изменения спроса на товары и оценки эффективности расположения товаров в торговых залах универсамов.

2. МОДЕЛИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

2.1. Каноническая форма. Прямая и двойственная формулировка линейных моделей

ММ линейного программирования отличаются следующими особенностями. Показатель эффективности W представляет собой линейную функцию от элементов решения (x_1, x_2, \dots, x_n) . Элементы решения представляют собой ряд неотрицательных переменных $x_i \geq 0, i = \overline{1, n}$. Ограничительные условия имеют вид линейных уравнений и неравенств. Ищутся такие решения, которые минимизируют или максимизируют W . Такие системы нельзя решать путём дифференцирования по аргументам x_i , так как экстремум достигается на границах области допустимых значений x_i , то есть там, где действуют ограничения. Чтобы обеспечить линейность ММ, необходимо выполнение допущения делимости и аддитивности. Первое допущение означает, что для каждого технологического процесса (ТПР) суммарное количество ресурсов и прибыль строго пропорциональны объёму выпускаемой продукции. Все показатели ТПР могут быть одновременно увеличены или уменьшены при сохранении их взаимной пропорциональности. Второе допущение показывает, что полное количество потребляемых ресурсов, затраченных при реализации ТПР, суммируется, а полная прибыль равна сумме прибылей в реализации этих ТПР.

Общая формулировка линейных моделей ИСО имеет вид

$$\begin{aligned} & \text{найти } \max W = f(x_1, x_2, \dots, x_n) \\ & \text{при ограничениях} \\ & G_j(x_1, x_2, \dots, x_n) \geq b_j, \quad j = \overline{1, n} \\ & x_i \geq 0, \quad i = \overline{1, n}. \end{aligned} \quad (2.1)$$

Существует несколько канонических форм ММ ЛП. Наиболее распространена запись ММ ЛП вида

$$\begin{aligned} & \max \sum_{i=1}^n c_i x_i; \\ & \sum_{i=1}^n a_{ij} x_i \leq b_j; \quad x_i \geq 0; \quad i = \overline{1, n}; \quad j = \overline{1, m}. \end{aligned} \quad (2.2)$$

Любую задачу линейного программирования можно рассматривать как задачу о распределении ограниченных ресурсов b_1, b_2, \dots, b_m между n потре-

бителями, представленных столбцами матрицы ограничений (a_{ij}) задачи. Любое допустимое решение задачи $(x_1^y, x_2^y, \dots, x_n^y)$ даёт конкретное распределение, указывая ту долю каждого из ресурсов, которая должна использоваться при осуществлении соответствующего ТПР.

Если в задаче при малом количестве переменных имеется большое число ограничений $(m > n)$, то переходят к формулировке двойственной задачи:

$$\min \sum_{i=1}^m b_i y_i; \quad (2.3)$$

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

В матричном виде пара двойственных задач записывается следующим образом:

$$\left. \begin{array}{l} \max C^T X; \\ AX \leq B; \\ X \geq 0; \end{array} \right\} \quad (2.4) \quad \left. \begin{array}{l} \min B^T Y; \\ A^T X \geq C; \\ Y \geq 0. \end{array} \right\} \quad (2.5)$$

Сопоставляя формы записи прямой и двойственной задач ЛП, можно установить следующие взаимосвязи:

- если прямая задача ЛП является задачей максимизации, то двойственная будет задачей минимизации, и наоборот;
- коэффициенты целевой функции прямой задачи c_1, c_2, \dots, c_n становятся свободными членами ограничений двойственной задачи;
- свободные члены ограничений прямой задачи b_1, b_2, \dots, b_m становятся коэффициентами целевой функции двойственной задачи;
- матрицу ограничений двойственной задачи получают транспонированием матрицы ограничений прямой задачи;
- знаки неравенств в ограничениях изменяются на обратные;
- число ограничений прямой задачи равно числу переменных двойственной задачи, а число ограничений двойственной задачи равно числу переменных прямой задачи.

Переменные y_1, y_2, \dots, y_m двойственной задачи иногда называют «теневыми» ценами. Для случая смешанных ограничений соотношения между прямой и двойственной задачами имеют вид

$$\left. \begin{array}{l} \max \sum_{j=1}^n c_j x_j; \\ \sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i=1, \dots, m_1 \leq m; \\ \sum_{j=1}^n a_{ij} x_j = b_i, \quad i=m_1+1, \dots, m; \\ x_j \geq 0, \quad j=1, 2, \dots, n_1 \leq n. \end{array} \right\} (2.6) \quad \left. \begin{array}{l} \min \sum_{i=1}^m b_i y_i; \\ \sum_{i=1}^m a_{ij} y_i \geq c_j, \quad j=1, \dots, n_1 \leq n; \\ \sum_{i=1}^m a_{ij} y_i = c_j, \quad j=n_1+1, \dots, n; \\ y_i \geq 0, \quad i=1, 2, \dots, m_1 \leq m. \end{array} \right\} (2.7)$$

Как видим, задача, сопряжённая с задачей со смешанными условиями, составляется согласно следующим правилам:

- если переменная x_j прямой задачи больше 0, то j -е условие системы ограничений двойственной задачи является неравенством;
- если на x_j не накладывается такое ограничение, то j -е ограничение двойственной задачи будет равенством.

Рассмотрим геометрический смысл задачи линейного программирования. Возможны четыре случая: задача имеет единственное оптимальное решение, существует множество оптимальных значений, случай неограниченно возрастающего оптимального решения, не имеет решения.

Пример 1. Единственное решение. **Пример 2.** Множество решений.

$$\left. \begin{array}{l} \max 12x_1 + 15x_2; \\ 4x_1 + 3x_2 \leq 12; (1) \\ 2x_1 + 5x_2 \leq 10; (2) \\ x_1 \geq 0; \quad x_2 \geq 0. \end{array} \right\} \quad \left. \begin{array}{l} \max 4x_1 + 10x_2; \\ 4x_1 + 3x_2 \leq 12; (1) \\ 2x_1 + 5x_2 \leq 10; (2) \\ x_1 \geq 0; \quad x_2 \geq 0. \end{array} \right\}$$

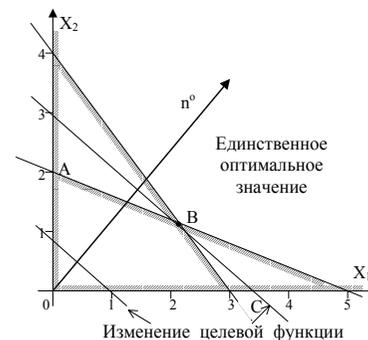


Рис. 2.1. Геометрический смысл примера 1

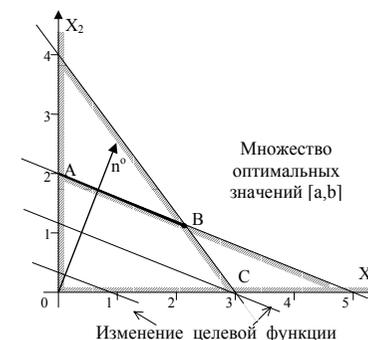


Рис. 2.2. Геометрический смысл примера 2

Область ограничений – выпуклый многоугольник OABC – одна и та же для обоих примеров. Различие – в целевых функциях. В первом случае целевая функция возрастает по нормали n_1^0 (прямые (а), (б) и (в) – разные значения целевой функции). В точке B имеет место единственное оптимальное решение. Во втором случае целевая функция возрастает по нормали n_2^0 (прямые (а), (б) и (в) также означают разные значения целевой функции). Вдоль отрезка АВ прямая (в) совпадает с линией ограничений (2). Поэтому задача имеет бесконечное множество решений, совпадающих с отрезком АВ.

Пример 3. Неограниченно возрастающее оптимальное решение.

$$\left. \begin{aligned} \max \quad & -2x_1 + 6x_2; \\ & -1x_1 - 1x_2 \leq -2; \quad (1) \\ & -1x_1 + 1x_2 \leq 1; \quad (2) \\ & x_1 \geq 0; \quad x_2 \geq 0. \end{aligned} \right\}$$

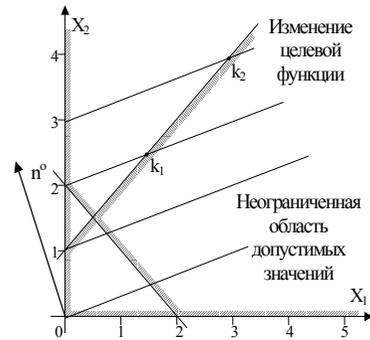


Рис. 2.3. Геометрический смысл примера 3

Пример 4. Не имеет решения.

$$\left. \begin{aligned} \max \quad & 1x_1 + 1x_2; \\ & -1x_1 + 1x_2 \leq -1; \quad (1) \\ & 1x_1 - 1x_2 \leq -1; \quad (2) \\ & x_1 \geq 0; \quad x_2 \geq 0. \end{aligned} \right\}$$

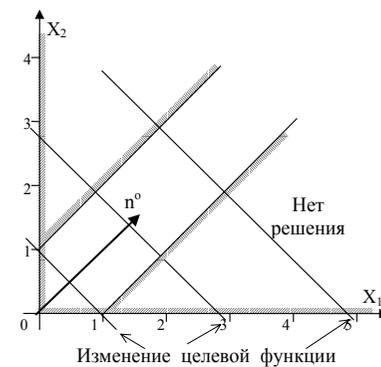


Рис. 2.4. Геометрический смысл примера 4

В третьем случае целевая функция возрастает по нормали n_3^0 (прямые (а), (б) и (в)). Каждый раз на прямой (1) можно найти точки k_1 и k_2 такие, что следующее значение W будет больше предыдущего и при этом будут удовлетворяться все ограничения задачи. Мы имеем случай неограниченного роста оптимального решения из-за того, что область допустимых решений не ограничена справа. В четвертом случае область допустимых решений распадается на две несовместимые части. Поэтому задача не имеет решения.

2.2. Примеры составления линейных моделей

Формулировка задач линейного программирования в каждом конкретном случае имеет свои особенности. Однако, можно рекомендовать придерживаться следующей технологии составления линейных моделей. Получение канонической формы возможно при последовательной реализации следующих этапов.

Этап 1. Составление содержательного описания ТПР. Первым шагом этапа 1 является определение состава управляемых переменных x_j . Устанавливается диапазон их изменения (x_{minj}, x_{maxj}). Проверяется выполнимость условия $x_j \geq 0$. На втором шаге определяется состав ресурсов, используемых при реализации ТПР. Определяется список ограничений на выбор решения (\bar{X}_p). Составляется содержательная формулировка показателя эффективности W моделей ЛП. Уточняются размеры дохода от реализации ТПР.

Этап 2. Табличная или матричная формулировка содержательного описания. Строками определяются виды ограничений. Отдельными строками записываются векторы доходов с единицы продукции и вектор объемов выпуска продукции (X_j). Столбцами являются различные варианты ТПР. Отдельным столбцом задается вектор наличных ресурсов, которым должны соответствовать ограничения.

Этап 3. Перевод табличной или матричной формулировки задачи в каноническую форму.

Для иллюстрации применения приведенной методики рассмотрим несколько примеров.

Пример 1. Классическая транспортная задача. Пусть имеется m предприятий, добывающих ресурсы. Производительность каждого из предприятий составляет соответственно a_1, a_2, \dots, a_m . Имеется n потребителей этих ресурсов. Спрос на ресурсы каждым из j -х потребителей соответствует b_1, b_2, \dots, b_n . Возможно перемещение любого ресурса от i -го производителя к j -му потребителю. При этом стоимость перевозки единицы ресурса составляет c_{ij} . Необходимо определить матрицу перевозок ресурса от i -го производителя к j -му потребителю $\|\bar{X}_{ij}\|$ такую, что стоимость затрат на перевозки

минимальна. Этап 2 ввиду простоты задачи можно пропустить. Поэтому можно сразу перейти к формализации процесса перевозок ресурса. Составим

целевую функцию $W(\bar{X}) = \min \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij}$ при ограничениях на произ-

$$\text{водство ресурсов } \sum_{j=1}^m x_{ij} = a_i; \quad i = \overline{1, m}; \quad x_{ij} \geq 0. \quad (a)$$

$$\text{Ограничения на потребление ресурсов } \sum_{i=1}^m x_{ij} = b_j; \quad j = \overline{1, n}. \quad (6)$$

Наконец, необходимо предположить для совместности уравнений (а) и (б) выполнимость условия $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j = A$.

Это условие означает, что общая мощность поставщиков ресурсов равна общей мощности потребителей. Общий объём ресурса равен A .

Пример 2. Классическая задача распределения ресурсов.

Этап 1. Содержательная постановка задачи. Предприятие выпускает неоднородную продукцию. Руководитель хочет определить уровни производства продукции по четырём ТПР (x_1, x_2, x_3, x_4). Доход от выпуска единицы продукции по i -му ТПР составляет c_i . При этом выпуске продукции предприятие несёт следующие затраты:

- трудозатраты (общим числом A единиц);
- расход материала 1 (общим числом B единиц);
- расход материала 2 (общим числом D единиц);

Этап 2. Составим таблицу, в которую занесём содержательную информацию о предприятии.

| Характеристики | Размерность | На единицу продукции 1-го типа | | На единицу продукции 2-го типа | | Имеется в наличии |
|---------------------------|-------------|--------------------------------|------------------|--------------------------------|------------------|-------------------|
| | | ТПР ₁ | ТПР ₂ | ТПР ₃ | ТПР ₄ | |
| Трудозатраты | ед. | a_1 | a_2 | a_3 | a_4 | A |
| Расход материала 1 | ед. | b_1 | b_2 | b_3 | b_4 | B |
| Расход материала 2 | ед. | d_1 | d_2 | d_3 | d_4 | D |
| Доход с единицы продукции | руб. | c_1 | c_2 | c_3 | c_4 | max |
| Объём выпуска продукции | ед. | x_1 | x_2 | x_3 | x_4 | найти |

Этап 3. Перевод табличной формы задачи в каноническую. Необходимо найти вектор, четырьмя компонентами которого будут объёмы выпуска продукции по ТПР _{i} $W = \max \sum_{i=1}^4 c_i x_i$ при ограничениях $\sum_{i=1}^4 a_i x_i \leq A$;

$$\sum_{i=1}^4 b_i x_i \leq B; \quad \sum_{i=1}^4 d_i x_i \leq D; \quad x_i \geq 0; \quad i = \overline{1, 4}.$$

Пример 3. Задача составления смесей жидкостей.

Этап 1. Содержательная постановка задачи. Предприятие торгует смесями жидкостей и выпускает три вида продуктов по трём технологическим процессам. Отличие ТПР _{i} ($i = \overline{1, 3}$) состоит в разных комбинациях двух составных компонент. У предприятия имеется всего в наличии S_1 количества

компоненты 1 и S_2 количества компоненты 2. Спрос на каждый вид продукции различен и составляет соответственно D_1, D_2 и D_3 . Однако спрос может быть удовлетворён при наличии сертификата каждого продукта. Так, для продукта, полученного по ТПР₁, спрос определяется концентрацией составляющих a_1/r_1 для первой компоненты и a_2/r_1 – для второй. Для продукта, полученного по ТПР₂, спрос определяется концентрацией составляющих b_1/r_2 для первой компоненты и b_2/r_2 – для второй. Для продукта, полученного по ТПР₃, спрос определяется концентрацией второй составляющей $1/r_3$. Доход, получаемый от продажи продуктов, полученных по разным технологиям, различен и составляет P_{ij} (где $i = 1, 2$ – номер компоненты в продукте, $j = 1, 2, 3$ – номер ТПР). Требуется найти уровень производства x_{ij} каждой компоненты в j -м ТПР.

Этап 2. Табличная формулировка содержательного описания. Составим таблицу, в которой будут следующие группы строк: наличие химической компоненты, спрос, технологии, доход, уровень производства.

| Условия | Химическая компонента 1 в продукте, полученном по | | | Химическая компонента 2 в продукте, полученном по | | | Ограничения |
|-------------------------------|---|------------------|------------------|---|------------------|------------------|-------------|
| | ТПР ₁ | ТПР ₂ | ТПР ₃ | ТПР ₁ | ТПР ₂ | ТПР ₃ | |
| Наличие компоненты 1 | 1 | 1 | 1 | | | | $\leq S_1$ |
| Наличие компоненты 2 | | | | 1 | 1 | 1 | $\leq S_2$ |
| Спрос на продукцию 1 | 1 | | | 1 | | | $\geq D_1$ |
| Спрос на продукцию 2 | | 1 | | | 1 | | $\geq D_2$ |
| Спрос на продукцию 3 | | | 1 | | | 1 | $\geq D_3$ |
| Технологические ограничения 1 | a_1/r_1 | | | a_2/r_1 | | | $\geq D_1$ |
| Технологические ограничения 2 | | b_1/r_2 | | | b_2/r_2 | | $\geq D_2$ |
| Технологические ограничения 3 | | | | | | $1/r_3$ | ≥ 0 |
| Доход | P_{11} | P_{12} | P_{13} | P_{21} | P_{22} | P_{23} | max |
| Уровень производства | x_{11} | x_{12} | x_{13} | x_{21} | x_{22} | x_{23} | найти |

Этап 3. Перевод табличной формы в каноническую. Необходимо найти матрицу $|x_{ij}|$ уровней производства по разным ТПР _{i} , дающую максимум

$$\text{целевой функции } W = \max \sum_{i=1}^2 \sum_{j=1}^3 P_{ij} x_{ij} \text{ при ограничениях на объёмы по}$$

требляемых химических компонент $x_{11} + x_{12} + x_{13} \leq S_1; \quad x_{21} + x_{22} + x_{23} \leq S_2$.

Запланированные выпуски продукции должны удовлетворять имеющемуся спросу $x_{11} + x_{21} \geq D_1; \quad x_{12} + x_{22} \geq D_2; \quad x_{13} + x_{23} \geq D_3$.

Технологические ограничения на приготовление смеси будут иметь следующий вид.

$$\text{На продукт, полученный по ТПР}_1: \frac{a_1}{r_1} x_{11} + \frac{a_2}{r_1} x_{21} \geq D_1 = x_{11} + x_{21}.$$

После преобразования имеем $\frac{a_1 x_{11} + a_2 x_{21}}{x_{11} + x_{21}} \geq r_1$.

На продукт, полученный по ТПП₂: $\frac{b_1}{r_2} x_{12} + \frac{b_2}{r_2} x_{22} \geq D_2 = x_{12} + x_{22}$.

После преобразования имеем $\frac{b_1 x_{12} + b_2 x_{22}}{x_{12} + x_{22}} \geq r_2$.

На продукт, полученный по ТПП₃: $\frac{x_{13}}{x_{23}} \geq r_3$.

Наконец, последняя группа ограничений $x_{ij} \geq 0$; $i = \overline{1, 2}$; $j = \overline{1, 3}$.

Пример 4. Задача выбора кратчайшего пути.

Этап 1. Содержательная формулировка. Имеется сеть, состоящая из восьми городов. Города связаны друг с другом сетью дорог. Известна стоимость c_{ij} проезда на автомобиле по дороге, связывающей i -й город с j -м. В некоторых случаях вместо стоимости можно рассматривать либо длину дороги l_{ij} , либо время проезда t_{ij} . На некоторых участках возможны циклы, означающие допустимость проезда в обоих направлениях. В сети выделены город, из которого мы начинаем движение (исток), и город, до которого необходимо добраться (сток). Необходимо выбрать либо самый дешёвый маршрут, либо кратчайший путь от источника до стока. В распоряжение исследователя предоставлен граф соединения городов сетью дорог, на котором указаны стоимости проезда по дорогам (рис. 2.5).

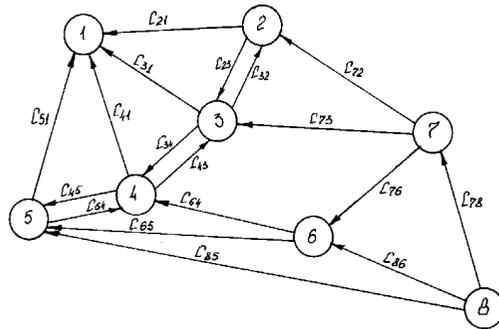


Рис. 2.5. Граф соединения городов сетью дорог

Этап 2. Матричная формулировка содержательного описания. Обозначим $c_{ij} = \infty$ в тех случаях, когда узел i не связан с узлом j . Введём переменную

$$x_{ij} = \begin{cases} 1, & \text{если узел } i \text{ соединён с узлом } j; \\ 0, & \text{в противном случае.} \end{cases}$$

Составим матрицу стоимости проезда $\|c_{ij}\|$ из i -го города к j -му городу

(узлы в сети):

| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Узел i |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| 8 | c_{87} | c_{86} | c_{85} | ∞ | ∞ | ∞ | ∞ | |
| 7 | 0 | c_{76} | ∞ | ∞ | c_{73} | c_{72} | ∞ | |
| 6 | ∞ | 0 | c_{65} | c_{64} | ∞ | ∞ | ∞ | |
| 5 | ∞ | ∞ | 0 | c_{54} | ∞ | ∞ | c_{51} | |
| 4 | ∞ | ∞ | c_{45} | 0 | c_{43} | ∞ | c_{41} | |
| 3 | ∞ | ∞ | ∞ | c_{34} | 0 | c_{32} | c_{31} | |
| 2 | ∞ | ∞ | ∞ | ∞ | c_{23} | 0 | c_{21} | |
| Узел j | | | | | | | | |

Этап 3. Перевод матричной формы в каноническую. Целевая функция задачи имеет вид $W = \min \sum_{i=1}^7 \sum_{j=2}^8 x_{ij} c_{ij}$.

Систему ограничений будем формировать по одному уравнению для каждого узла и использовать по одной переменной для каждой дуги. Условие выезда из источника будет иметь вид $x_{85} + x_{86} + x_{87} = 1$.

Для каждого промежуточного узла сумма входов равна сумме выходов (нет накопления в узле). Поэтому надо приравнять суммы входов (положительные по строке) и разности выходов (отрицательные по столбцам):

$$\begin{aligned} x_{72} + x_{73} + x_{76} - x_{87} &= 0; & x_{64} + x_{65} - x_{76} - x_{86} &= 0; \\ x_{54} - x_{51} - x_{85} - x_{65} - x_{45} &= 0; & x_{45} + x_{43} + x_{41} - x_{64} - x_{54} - x_{34} &= 0; \\ x_{34} + x_{32} + x_{31} - x_{73} - x_{43} - x_{23} &= 0; & x_{23} + x_{21} - x_{51} - x_{41} - x_{31} - x_{21} &= 0. \end{aligned}$$

Условие прибытия на конечный пункт имеет вид $x_{21} + x_{31} + x_{41} + x_{51} = -1$. Завершим формализацию задачи формулировкой самых главных ограничений: $x_{ij} \geq 0$; $i = \overline{1, 7}$; $j = \overline{2, 8}$.

Решение указанной системы уравнений, обеспечивающее минимум целевой функции W , и будет решением задачи выбора кратчайшего пути.

2.3. Симплекс-метод решения задачи линейного программирования

2.3.1. Общие определения

Симплекс-метод разработан в 1949 году американским математиком Ф. Данцигом для решения задач, записанных в векторной форме:

$$\left. \begin{aligned} \max W &= CX, \\ AX &= B, X \geq 0, \end{aligned} \right\} \quad (2.8)$$

где A – матрица размерности $m \times n$ условий задачи ранга m ; $C = (c_1, c_2, \dots, c_n)$ – вектор-строка коэффициентов целевой функции W ; $B^T = (b_1, b_2, \dots, b_m)$ – вектор-столбец коэффициентов ограничений; $X^T = (x_1, x_2, \dots, x_n)$ – вектор-столбец, удовлетворяющий ограничениям, называется допустимым решением или планом задачи ЛП. Допустимые решения, доставляющие целевой функции W максимальное значение, называются оптимальным решением (оптимальным планом). Множество векторов $Ax = B$ называется областью допустимых решений задачи ЛП и при этом является выпуклым. Поскольку в канонической форме задача ЛП формулируется в виде системы линейных алгебраических уравнений, а в симплекс-методе преобразования этих уравнений являются основной операцией, то приведём основную сводку правил и подходов к решению систем линейных уравнений. Рассмотрим следующий пример. Пусть имеется две системы уравнений:

$$\left. \begin{array}{l} 2x_1 - x_2 + 2x_3 + x_4 = 1; \\ x_1 + x_2 - x_3 + 3x_4 = 5; \\ 3x_1 - x_2 + 3x_3 - x_4 = 5; \end{array} \right\} \quad (2.9) \quad \left. \begin{array}{l} x_1 + 3x_4 = 1; \\ x_2 - 5x_4 = 7; \\ x_3 - 5x_4 = 3. \end{array} \right\} \quad (2.10)$$

В системе (2.10) каждое из уравнений содержит переменную (x_1, x_2, x_3) , которая входит в него с коэффициентом 1 и исключена из остальных уравнений (имеет коэффициент, равный нулю). Такие переменные называются **базисными**, а система (2.10) называется системой в базисной форме. Векторы-столбцы базисных переменных x_1, x_2 и x_3

$$a^{(1)} = \begin{vmatrix} 1 \\ 0 \\ 0 \end{vmatrix}; \quad a^{(2)} = \begin{vmatrix} 0 \\ 1 \\ 0 \end{vmatrix}; \quad a^{(3)} = \begin{vmatrix} 0 \\ 0 \\ 1 \end{vmatrix}. \quad (2.11)$$

Они и составляют базис системы (2.10). Базис системы (2.9) также может быть образован из вектор-столбцов при переменных x_1, x_2, x_3 , ибо определитель третьего порядка системы (2.9) отличен от нуля. Преимущество системы (2.10) в том, что её общее решение находится непосредственно через переменную x_4 . Таким образом, базисные (зависимые) переменные x_1, x_2, x_3 выражены через свободную (независимую) переменную x_4 . Например, при $x_4 = 0$ получим некоторые **частные** решения ($x_1 = 1; x_2 = 7; x_3 = 3$). Частное решение, в котором свободные неизвестные равны нулю, называются **базисными**. Вектор $X = (1, 7, 3, 0)$ также является решением системы (2.9). При этом матрица коэффициентов системы (2.10) в отличие от (2.9) содержит единичную подматрицу (2.11). Благодаря этому становится возможным непосредственное решение системы (2.10).

Из теории матриц известно, что единичная матрица есть результат умножения квадратной неособенной матрицы на обратную ей. Отсюда вытекает следующая схема решения системы линейных уравнений:

$$AX = B. \quad (2.12)$$

Разобьём матрицу A на две подматрицы Q и R так, что Q будет квадратной неособенной матрицей $m \times m$, а R – матрица размерности $m \times (n - m)$, состоящая из остальных компонентов матрицы A . Так как матрица Q неособенная, то её векторы-столбцы образуют базис системы векторов-столбцов матрицы A , и её назовём **базисной**. Аналогичным образом разобьём компоненты вектора X так, чтобы $X = (X_Q, X_R)$, где X_Q – вектор базисных переменных, X_R – вектор небазисных переменных. В результате исходная система может быть записана в виде

$$QX_Q + RX_R = B.$$

Для неё ищется общее решение в виде (X_Q, X_R) . Здесь X_R – произвольный $(n - m)$ -мерный вектор. X_Q выражается в виде

$$X_Q = Q^{-1}B - Q^{-1}RX_R.$$

Базисное решение $(X_Q, X_R) = (Q^{-1}B, 0)$, полученное при $X_R = 0$, единственно, так как Q – неособенная матрица. Если рассматривается система (2.12), то базисное решение $(Q^{-1}B, 0)$ и соответствующий ему базис Q называется допустимым при условии $Q^{-1}B \geq 0$. Базисное допустимое решение называется **невырожденным**, если все его компоненты **положительны**, и вырожденным в противном случае. Понятие базисного допустимого решения совпадает с понятием опорного плана.

2.3.2. Метод исключений Жордана-Гаусса

С вычислительной точки зрения схема решения, основанная на обращении базисной матрицы, может быть реализована различными способами. Метод исключений Жордана преобразует исходную систему

$$\sum_{j=1}^n a_{ij} x_j = b_i; \quad i = \overline{1, m} \quad (2.13)$$

в базисную

$$x_i + \sum_{j=m+1}^n a'_{ij} x_j = b'_i, \quad (2.14)$$

где коэффициенты a'_{ij} являются элементами матрицы $Q^{-1}R$ размером $m \times (n - m)$, а b'_i – элементами m -мерного вектора $Q^{-1}B$. Метод исключений Жордана позволяет решить систему преобразованием её к эквивалентной

системе с единичным базисом или установить её несовместность. В процессе преобразований, при необходимости, может быть построена также обратная матрица. Это многошаговая процедура, когда на каждом шаге исключений одна из переменных становится базисной и исключается из всех уравнений (кроме одного, где и остаётся с единичным коэффициентом). Идея алгоритма сводится к следующему. Пусть переменная x_s входит в r -е уравнение с коэффициентом $a_{rs} \neq 0$. Для того чтобы она стала базисной, разделим r -е уравнение на $a_{rs} \neq 0$ (получим x_s с единичным коэффициентом) и результат вычтем из каждого оставшегося уравнения $i = \overline{1, m}$, $i \neq r$, умножая его каждый раз на соответствующий коэффициент a_{is} (и таким образом получаем x_s с нулевым коэффициентом) в остальных уравнениях. Совокупность операций, составляющих шаг жордановых исключений, называется жордановским преобразованием, а коэффициент a_{rs} – разрешающим (ведущим) элементом, строка r и столбец s соответственно являются ведущими строкой и столбцом. Формулы для расчёта коэффициентов a'_{ij} и b'_i новой системы уравнений будут иметь вид

$$\left. \begin{aligned} a'_{rj} &= \frac{1}{a_{rs}} a_{rj}; & j = \overline{1, n}; \\ b'_r &= \frac{1}{a_{rs}} b_r; \end{aligned} \right\} (2.15) \quad \left. \begin{aligned} a'_{ij} &= a_{ij} - \frac{a_{is}}{a_{rs}} a_{rj}; \\ b'_i &= b_i - \frac{a_{is}}{a_{rs}} b_r; \\ & j = \overline{1, n}; \quad i = \overline{1, m}; \quad i \neq r. \end{aligned} \right\} (2.16)$$

Вычисления по формулам (2.16) можно описать с помощью правила прямоугольника:

$$\begin{array}{cccc} a_{ij} & \cdots & a_{is} & \\ \vdots & & \vdots & \\ a_{ir} & \cdots & a_{rs} & \end{array}$$

Чтобы найти преобразованный элемент a'_{ij} , из элемента a_{ij} надо вычесть произведение коэффициентов, стоящих напротив его в ведущих столбце и строке, и поделить на ведущий элемент, расположенный по диагонали от элемента a_{ij} . При этом элементы ведущего столбца принимают значения $a'_{ij} = 0$ для всех $i = \overline{1, m}$, $i \neq r$, $a'_{rs} = 1$.

Итак, последовательность действий, выполняемых на одном шаге жордановских исключений, следующая:

– ведущий элемент заменяется на единицу;

- все остальные элементы ведущей строки делятся на ведущий элемент;
- все остальные элементы ведущего столбца заменяются нулями;
- элементы, не принадлежащие ведущей строке или столбцу, вычисляются по правилу прямоугольника.

Для преобразования системы (2.13) в систему (2.14) требуется не более m шагов жордановских исключений. На первом шаге в качестве ведущего элемента выбирается любой элемент $a_{rs} \neq 0$. На втором шаге ведущий элемент выбирается в любом уравнении (кроме r -го) среди ненулевых элементов системы, полученной в результате первого шага. Далее процесс исключений повторяется. Если в процессе исключений появится уравнение, в котором левая часть равна нулю, а свободный член отличен от нуля, то это **признак несовместности системы** уравнений. Когда же и левая, и правая части уравнения равны нулю, то это означает, что данное уравнение является **линейной комбинацией** остальных уравнений и его следует исключить из рассмотрения. Таким образом, в процессе жордановских исключений либо устанавливается несовместность системы уравнений, либо система приводится к эквивалентной базисной форме вида (2.14), из которой решение получается непосредственным образом. Формулы жордановских преобразований (2.15) и (2.16) применяются одинаково как в случае $m = n$, так и при $m < n$.

2.3.3. Алгебра симплекс-метода

Основу вычислительной схемы симплекс-метода составляют преобразования базисных систем линейных уравнений. Каждое преобразование связано с переводом одной базисной переменной в список небазисных, а одной небазисной переменной – в список базисных. Замена переменных организуется таким образом, что целевая функция с каждым преобразованием возрастает, если только получаемые допустимые базисные решения (опорные планы) невырождены.

Рассмотрим задачу (2.8) в предположении, что система $AX = B$ приведена к эквивалентной базисной форме (2.14). Пусть для простоты записи базисными являются первые m переменных. Тогда общие и базисные решения имеют вид

$$x_i = b_i - \sum_{j=m+1}^n a_{ij} x_j; \quad i = \overline{1, m}; \quad (2.17)$$

$$x_i = \begin{cases} b_i, & \text{если } i = \overline{1, m}; \\ 0, & \text{если } i = \overline{m+1, n}. \end{cases} \quad (2.18)$$

Целевую функцию $W = CX$ преобразуем так, чтобы она зависела только от свободных переменных. Подставим для этого в неё выражение базисных

переменных из равенства (2.17) и получим

$$W = \sum_{i=1}^m c_i \left(b_i - \sum_{j=m+1}^n a_{ij} x_j \right) + \sum_{j=m+1}^n c_j x_j.$$

Обозначим

$$Z_0 = \sum_{i=1}^m c_i b_i, \quad (2.19) \quad Z_j = \sum_{i=1}^m c_i a_{ij}; \quad j = \overline{1, m}. \quad (2.20)$$

Приведём целевую функцию к виду

$$W = Z_0 - \sum_{j=m+1}^n (Z_j - c_j) x_j, \quad (2.21)$$

$$W = Z_0 - \sum_{j=m+1}^n \Delta_j x_j, \quad (2.22)$$

$$\Delta_j = Z_j - c_j; \quad j = \overline{1, n}. \quad (2.23)$$

Равенство (2.22) называется приведённым выражением целевой функции, а коэффициенты Δ_j – приведёнными коэффициентами целевой функции (или **оценками** соответствующих переменных x_j). Каждая базисная переменная входит в одну строку системы с коэффициентом 1, а в остальные строки – с коэффициентом 0. Из уравнения (2.20) следует, что для базисных переменных имеет место равенство $z_j = c_j$, откуда получаем, что оценки базисных переменных равны нулю ($\Delta_j = Z_j - c_j = 0$, $j = \overline{1, m}$). Проанализируем связь между величиной целевой функции и свободными переменными. Возможны три случая.

Случай 1. Все коэффициенты Δ_j при свободных переменных x_j неотрицательны ($\Delta_j \geq 0$, $j = m+1, \dots, n$). Тогда любое увеличение свободных переменных $x_{m+1}, x_{m+2}, \dots, x_n$, которые в базисном решении равны нулю, привело бы к уменьшению функции (2.22). Следовательно, допустимое базисное решение (2.18) даёт максимум целевой функции ($\max W = Z_0$) и является оптимальным. Итак, условие $\Delta_j = Z_j - c_j \geq 0$, $j = \overline{1, n}$ является условием оптимальности плана задачи.

Случай 2. Среди оценок Δ_j свободных переменных имеется хотя бы одна отрицательная и при этом среди коэффициентов соответствующего ей столбца базисной системы нет положительных. В таких случаях целевая функция W не ограничена в области допустимых решений. В самом деле, пусть Δ_s и все $a_{is} \leq 0$, $i = \overline{1, m}$. Тогда, полагая в (2.17) и (2.22), что все сво-

бодные переменные, кроме x_s , равны нулю, получим

$$x_i = b_i - a_{is} x_s; \quad i = \overline{1, m}; \quad (2.24)$$

$$Z = Z_0 - \Delta_s X_s. \quad (2.25)$$

Так как $\Delta_s < 0$, то при увеличении x_s функция (2.25) монотонно возрастает. Вместе с тем, из (2.24) следует, что при $a_{is} < 0$ все переменные x_i , зависящие от x_s , остаются отрицательными при любых, сколь угодно больших, x_s . Здесь мы имеем случай 3 из геометрической интерпретации моделей ЛП. Таким образом, функция W максимума не имеет. Оптимального плана у задачи нет, хотя допустимых планов может быть бесконечно много и W всё время монотонно растёт с ростом x_s .

Случай 3. Имеются свободные переменные x_s с оценкой $\Delta_s < 0$ и хотя бы один коэффициент $a_{is} < 0$. Тогда можно найти новый опорный план, для которого значение целевой функции больше, если рассматривается невырожденный опорный план. В самом деле, как видно из (2.25), при $\Delta_s < 0$ увеличение x_s приводит к росту целевой функции Z . При этом увеличение x_s допустимо до тех пор, пока правые части выражения (2.24) остаются неотрицательными при всех i . Максимально возможное значение $X_s = \Theta_s$, при котором обеспечивается неотрицательность переменных в (2.24), определяется из условия

$$\Theta_s = \min_{a_{is} > 0} \frac{b_i}{a_{is}} = \frac{b_r}{a_{rs}}. \quad (2.26)$$

Для невырожденного опорного плана все $b_i > 0$. Следовательно, $\Theta_s > 0$. Присваивая в соответствии с (2.26) переменной X_s значение

$$X_s = \Theta_s = \frac{b_r}{a_{rs}}, \quad (2.27)$$

мы переводим переменную X_s в число базисных вместо базисной переменной X_r .

Для определения значений остальных базисных переменных x_i , $i = \overline{1, m}$, $i \neq s$ по формулам (2.16) проводятся жордановы исключения с ведущим элементом a_{rs} . В итоге базисное решение преобразованной системы примет вид

$$\left. \begin{aligned} x_i &= b_i - \frac{a_{is} b_r}{a_{rs}}; \quad i = \overline{1, m}; \quad i \neq r; \\ x_r &= 0; \quad x_s = \frac{b_r}{a_{rs}}; \quad x_j = 0; \quad j = m+1, \dots, n; \quad j \neq s. \end{aligned} \right\} \quad (2.28)$$

Отметим, если $\min(b_i/a_{is})$ в (2.16) достигается в нескольких строках ($i = r, t, \dots$), то нулевые значения примут не одна, а несколько соответствующих переменных x_r, x_t, \dots . Новая базисная переменная x_s вводится вместо любой из них, а остальные остаются в базисе. Полученный в этом случае план является вырожденным.

Рассмотрим, как изменяется целевая функция при переходе к новому опорному плану. В соответствии с равенствами (2.25) и (2.27) новое значение целевой функции будет равно

$$z' = z_0 - \left(\Delta_s / a_{rs} \right) b_r. \quad (2.29)$$

Так как $\Delta_s < 0$ и $b_r/a_{rs} > 0$, то целевая функция возрастёт на величину $\Delta_s \Theta_s = \Delta_s (b_r/a_{rs})$. Как видим, выражение (2.29) повторяет формулу (2.26) жордановых исключений для расчёта правых частей системы (2.14) с заменой коэффициента a_{is} строки i матрицы A на коэффициент Δ_s приведённой целевой функции. Действительно, выражение (2.19), записанное в виде

$$z + \sum_{j=m+1}^n \Delta_j x_j = z_0, \quad (2.30)$$

формально не отличается от уравнений системы (2.14). Поэтому (2.30) можно рассматривать как $(m+1)$ -е уравнение, разрешённое относительно базисной переменной z , и выполнять над элементами Δ_j и z_0 те же преобразования, что и над элементами a_{ij}, b_i . Расчетная формула имеет вид

$$\Delta'_j = \Delta_j - \left(\frac{\Delta_s}{a_{rs}} \right) a_{rj}; \quad j = \overline{1, n}. \quad (2.31)$$

Итак, выполнив жорданово преобразование системы (2.14), расширенной добавлением строки приведённой целевой функции (2.29), получим новое базисное допустимое решение системы (2.14), связанное с большим значением целевой функции. Чтобы решить, является ли новое базисное решение оптимальным, надо провести анализ оценок Δ_j . При этом возникает опять один из трёх взаимоисключающих случаев. Если имеет место случай 3, то процесс жордановых преобразований продолжается.

После конечного числа преобразований при невырожденном начальном опорном плане и невырожденных опорных планах на каждой итерации возможен случай 1 или 2. То есть либо будет найдено оптимальное решение, либо будет установлена неограниченность целевой функции. Отметим, что конечность симплекс-метода является следствием конечности числа различных опорных планов (с верхней границей C_n^m). Если в процессе работы алгоритма появляется вырожденный опорный план, то на этой операции пере-

ход к следующему опорному плану происходит без увеличения целевой функции.

2.3.4. Вычислительная схема симплекс-метода для задачи в базисной форме

Задачей в базисной форме называется каноническая задача линейного программирования

$$\max \{ W = C X \mid A X = B, \quad X > 0 \}$$

с m уравнениями, n неизвестными и неотрицательным вектором B , разрешённая относительно m переменных, имеющая единичный базис. Условие такой задачи удобно задавать в виде симплекс-таблицы

| C_Q | X_Q | B | c_1 a^1 | c_2 a^2 | \dots \dots | c_n a^n |
|----------|----------|----------|----------------|----------------|--------------------|----------------|
| c_1 | x_1 | B_1 | a_{11} | a_{12} | \dots | a_{1n} |
| c_2 | x_2 | B_2 | a_{21} | a_{22} | \dots | a_{2n} |
| \vdots | \vdots | \vdots | \vdots | \vdots | \vdots | \vdots |
| c_m | x_m | B_m | a_{m1} | a_{m2} | \dots | a_{mn} |
| | Δ | Z_0 | Δ_1 | Δ_2 | \dots | Δ_n |

В процессе решения в столбце X_Q находится список базисных переменных, а в столбце C_Q содержатся соответствующие им коэффициенты целевой функции. В столбце B находятся значения базисных переменных, а в столбцах a^1, a^2, \dots, a^n – коэффициенты a_{ij} матрицы условий. В оценочной строке Δ содержатся значение целевой функции и приведённые коэффициенты Δ_j при неизвестных линейной формы. При заполнении элементов Δ -строки исходной таблицы используются формулы (2.19), (2.20) и (2.23), при помощи которых целевая функция приводится к базису, в котором представлена система ограничений. Если целевая функция выражена только через свободные переменные (уже приведена к единичному базису), а коэффициенты при базисных переменных равны нулю, то коэффициенты Δ -строки определяются по формуле $\Delta_j = -c_j$ (поскольку $z_0 = 0, \Delta_j = 0$ для базисных переменных).

2.3.5. Алгоритм симплекс-метода

Алгоритм симплекс-метода реализуется в такой последовательности шагов.

Шаг 1. Просматриваем знаки коэффициентов Δ_j оценочной строки. Если все $\Delta_j \geq 0$, то задача решена: допустимое базисное решение (2.18) оптимально и $\max W = Z_0$. Если не все $\Delta_j \geq 0$, то переходим к шагу 2.

Шаг 2. Среди значений $\Delta_j < 0$ находим наибольшее по абсолютной величине и соответствующий ему столбец выбираем в качестве ведущего. Пусть это будет столбец с номером s . Если в ведущем столбце все элементы $a_{is} \leq 0$

для всех $i = \overline{1, m}$, то имеем случай 2, когда целевая функция неограничена ($\max W = \infty$), и решение также окончено. Если же не все $a_{is} \leq 0$, то переходим к шагу 3.

Шаг 3. Для каждого элемента $a_{is} > 0$ ведущего столбца находим отношение b_i / a_{is} , выбираем из них наименьшее и называем строку (где это имеет место) ведущей. Пусть это будет строка с номером r . Элемент a_{rs} (на пересечении ведущего столбца с ведущей строкой) будет ведущим элементом.

Шаг 4. Выполняем жорданово преобразование таблицы с ведущим элементом a_{rs} по формулам (2.15), (2.16), (2.31) и переходим к шагу 1.

Последовательность операций на шагах 1–4 называется итерацией симплекс-метода.

2.3.6. Построение начального опорного плана

Вычислительная схема симплекс-метода применима лишь к задачам в базисной форме, то есть к задачам, для которых известны начальный опорный план и соответствующий ему базис. До сих пор не рассматривалось, каким образом опорный план получается, поскольку его нахождение – это дополнительная задача. Если матрица A содержит единичную матрицу порядка m и вектор ограничений $B \geq 0$, то столбцы этой подматрицы составляют допустимый базис задачи ЛП. Переход от системы неравенств стандартной задачи

$$\max W = \sum_{j=1}^n c_j x_j; \quad (2.32)$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i; \quad i = \overline{1, m}; \quad (2.33)$$

$$x_j \geq 0; \quad j = \overline{1, n} \quad (2.34)$$

к эквивалентной системе уравнений канонической задачи

$$\max W = \sum_{j=1}^n c_j x_j; \quad \sum_{j=1}^n a_{ij} x_j + x_{n+i} = b_i, \quad i = \overline{1, m}; \quad x_j \geq 0; \quad j = \overline{1, n+m}$$

осуществляется введением дополнительных неотрицательных переменных $x_{n+i} \geq 0$, $i = \overline{1, m}$, которые входят в целевую функцию с нулевыми коэффициентами. В качестве начального опорного плана такой задачи можно рассматривать план $x_j = 0$; $j = \overline{1, n}$; $x_{n+i} = b_i$; $i = \overline{1, m}$.

Этот план является допустимым (удовлетворяет приведённым ограничениям). Итак, план $X = (0, 0, \dots, 0, b_1, b_2, \dots, b_m)$ будет опорным, и с него можно начать симплекс-процедуру поиска оптимального плана. Однако в ряде случаев вычислительная схема займёт очень много итераций. Поэтому существ-

вуют другие способы задания опорного плана. Рассмотрим два из наиболее распространённых методов.

Метод минимизации невязок. Для отыскания опорного плана канонической задачи (2.32)–(2.34) в общем случае может быть использовано решение следующей вспомогательной задачи ЛП с $(n+m)$ переменными

$$\max W = - \sum_{i=1}^m x_i; \quad (2.35)$$

$$\sum_{j=1}^n a_{ij} x_j + x_{n+i} = b_i; \quad i = \overline{1, m}; \quad (2.36)$$

$$x_j \geq 0; \quad j = \overline{1, n+m} \quad (2.37)$$

Она получена путём введения искусственных переменных (или невязок) $x_{n+i} \geq 0$, $i = \overline{1, m}$ с требованием минимизации их суммы. Для решения этой вспомогательной задачи уже можно применять симплекс-метод, так как имеется исходный опорный план: $x_j = 0$ при $j = \overline{1, n}$; $x_{n+i} = b_i$ при $i = \overline{1, m}$.

Поскольку целевая функция вспомогательной задачи ограничена сверху числом 0, то случай 2 (неограниченного возрастания решения) сразу же исключается. В итоге через конечное число итераций симплекс-метода возможны два исхода: либо оптимальное решение задачи (2.35)–(2.37) равно нулю (если максимум $W = 0$, то исходная задача (2.32)–(2.34) допустима), либо оптимальное значение целевой функции $W < 0$, что означает недопустимость исходной задачи (2.32)–(2.34). Таким образом, решение вспомогательной задачи (2.35)–(2.37) или приводит к некоторому опорному плану задачи (2.32)–(2.34), или устанавливает недопустимость исходной задачи. При этом симплекс-таблица, получаемая на последней итерации решения вспомогательной задачи, может быть использована при решении исходной задачи. Для этого надо отбросить столбцы, соответствующие искусственным переменным $x_{n+1}, x_{n+2}, \dots, x_{n+m}$, и заменить нулевые элементы оценочной строки элементами, которые вычисляются по формулам (2.20) и (2.23). Таким образом, процесс решения задачи ЛП разбивается на два этапа:

- определение опорного плана путём решения вспомогательной задачи;
- результат первого этапа затем используется для отыскания уже собственно оптимального плана задачи ЛП.

Метод искусственного базиса. Оба этапа объединяются. Вместо исходной канонической задачи (2.32)–(2.34) рассматривается расширенная задача с $(n+m)$ переменными, полученная введением в систему ограничений (2.33)

искусственных переменных $x_{n+i} \geq 0$, $i = \overline{1, m}$. Эти переменные включаются в целевую функцию с одинаковыми отрицательными коэффициентами $-M$, где $M > 0$ (сколь угодно большое число).

$$\max W = \sum_{j=1}^n c_j x_j - M \sum_{i=1}^m x_{n+i} \quad (2.38)$$

$$\sum_{j=1}^n a_{ij} x_j + x_{n+i} = b_i; \quad i = \overline{1, m} \quad (2.39)$$

$$x_j \geq 0; \quad j = \overline{1, n+m}. \quad (2.40)$$

Расширенная задача (2.38)–(2.40) называется M -задачей, а метод искусственного базиса – M -методом. Единичные вектор-столбцы $a_{n+1}, a_{n+2}, \dots, a_{n+m}$ матрицы условий M -задачи, соответствующие искусственным переменным, составляют искусственный единичный базис. Ему соответствует очевидный начальный опорный план $X = (0, 0, \dots, 0, b_1, b_2, \dots, b_n)$, где нулевыми являются первые n координат и значение целевой функции принимает вид

$$Z = -M \sum_{i=1}^m x_{n+i}.$$

Введение в целевую функцию коэффициентов $-M$ при искусственных переменных эквивалентно введению штрафа за включение в опорный план переменных x_{n+i} , $i = \overline{1, m}$. Обычно числа $-M$ по абсолютной величине значительно превосходят остальные коэффициенты и позволяют выводить из базиса искусственные переменные и вводить в базис переменные исходной задачи. Имея начальный опорный план, можно применить симплекс-метод для отыскания оптимального плана расширенной задачи (2.38)–(2.40).

2.4. Анализ линейных моделей на чувствительность

Любую задачу ЛП с экономической точки зрения можно рассматривать как задачу о распределении ограниченных ресурсов b_1, b_2, \dots, b_m между различными потребителями. Например, между некоторыми $ТПР_i$, которые представлены столбцами a^1, a^2, \dots, a^n матрицы ограничений задачи. Любое допустимое решение задачи ЛП (x_1, x_2, \dots, x_n) даёт конкретное распределение, указывая ту долю каждого из ресурсов, которая должна быть использована при осуществлении соответствующего $ТПР_i$. Если в задаче при малом количестве переменных имеется большое количество ограничений ($m > n$), то переходят к формулировке двойственной задачи.

Одновременно с желанием определить оптимальные значения для каждого из решений x_j у руководителя может возникнуть желание выяснить:

а) каким образом отразится на полученной прибыли увеличение каждого

из потребляемых ресурсов;

б) каким способом необходимо совершенствовать технологические процессы для роста прибыли;

в) возможно ли изменение стоимости сырья за счёт замены его менее дефицитным.

Подобного рода вопросы зачастую оказываются для руководителя предприятия более важными, чем определение оптимальных значений x_j . Решение этих проблем и называют **анализом на чувствительность** линейной модели. Теория двойственности задач ЛП позволяет проводить анализ моделей ЛП на чувствительность.

Если задана прямая задача $\max W(X) = \sum_{j=1}^n c_j x_j$ при условиях

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

то это означает, что требуется максимизировать доход $W(X)$ от реализации плана производства X при известной интенсивности расхода i -го ресурса при j -м способе $ТПР_i$ (a_{ij}) и известных уровнях i -го ресурса (b_i).

2.4.1. Варьирование ресурсов

Ресурсы заданы матрицей $B = \|b_i\|$. Ставится задача – найти ответы на вопросы:

– при каких вариациях правых частей ограничений оптимальный план x_0 не изменится;

– как эти вариации скажутся на величине дохода $W(X)$.

Если b_i получают вариации Δb_i , то это приводит к вариациям плана $x_0 \rightarrow x_0(b_0 + \Delta b)$ и целевой функции $W(x_0, b_0 + \Delta b)$. Пусть эти вариации Δb таковы, что план $x_0(b_0 + \Delta b)$ остаётся допустимым (т. е. удовлетворяет условию неотрицательности)

$$\Delta W_{\max}(b) = \frac{W(x_0, (b_0 + \Delta b)) - W(x_0, b_0)}{\Delta b}.$$

$$\text{Устремив } \Delta b \rightarrow 0, \text{ получим } \lim_{\Delta b_i \rightarrow 0} \frac{\Delta W_{\max}(b)}{\Delta b_i} = \frac{\partial W_{\max}(b)}{\partial b_i}.$$

Здесь b рассматривается как варьируемый параметр. Согласно теореме о двойственности $W_{\max}(x_0) = \sum_{j=1}^n c_j x_j^0 = \sum_{i=1}^m b_i y_i^*$, $\frac{\partial W_{\max}(b)}{\partial b_i} = y_i^*$; $i = \overline{1, m}$.

Таким образом, оптимальные значения двойственных переменных y_i^* определяют вклад каждого ресурса в доход W_{\max} при оптимальном значении

x_0 . Эта величина численно равна дополнительному доходу при увеличении i -го ресурса b_i на единицу и при условии, что ресурсы используются оптимальным образом. Итак, величины y_i^* служат показателями возможности i -го ресурса для системы. Чем больше значение y_i^* , тем существеннее вклад i -го ресурса в функцию дохода W_{max} и тем выгоднее его увеличение. Если же некоторое $y_i^* = 0$, то это означает, что i -й ресурс не является существенным ограничением для системы. Величины y_i^* двойственной задачи часто называют скрытыми доходами (маргинальными оценками ресурсов) системы. Они могут рассматриваться как потенциальная возможность получения дополнительной прибыли за счёт увеличения соответствующего ресурса при условии, что система функционирует нормально.

2.4.2. Варьирование целевой функции

Руководителя предприятия может интересовать возможность изменения коэффициентов целевой функции $\{c_j\}$; $j = \overline{1, n}$. Ставится вопрос: при каких условиях найденный ранее оптимальный план останется оптимальным и каковы при этом будут вариации коэффициентов c_j . Пусть c_j изменилось на величину $c_j^* = c_j + \delta_j$. Обозначим через I_B множество индексов, базисных векторов, а I_{NB} – множество индексов небазисных переменных в оптимальном решении x_0 . Найдём величину оценок после вариации для двух случаев:

$$1) \text{ если } r \in I_{NB}, \text{ то } \Delta_j^H = \Delta_j; j \neq r; \Delta_j^H = \sum_{i \in I_B} c_i a_{ir} - (c_j - \delta_j);$$

2) если $r \in I_B$, то

$$\Delta_j^H = \sum_{i \in I_B} c_i^H a_{ij} - (c_j - \delta_j) = \sum_{i \in I_B} c_i a_{ij} - \delta_j a_{jr} - c_j; j \in I_{NB}. \quad (2.41)$$

Для сохранения оптимальности прежнего плана при вариациях коэффициентов c_j необходимо и достаточно сохранение знаков оценки ($\Delta_j^H \geq 0$) для всех небазисных переменных. Поэтому из этого условия с помощью уравнений (2.41) можно определить допустимые вариации коэффициентов целевой функции (Δc_j), при которых сохраняется прежнее оптимальное решение.

Применение моделей ЛП при решении производственных задач сопряжено со следующими трудностями.

1. При формулировке ограничений оптимизации необходимо быть очень осторожным, чтобы не просмотреть важные ограничения. Но большое количество ограничений может привести к тому, что оптимизация вообще ока-

жется невозможной.

2. Используется ЭВМ в качестве инструмента организации итеративных вычислений. На практике число ограничений доходит до 200, а число неизвестных при этом в 3-4 раза превышает число ограничений.

3. Поскольку ММ ЛП представляют собой некоторую абстракцию, то нет полной гарантии, что первая же проба решения задачи приводит сразу же к успеху. И поэтому каждый раз необходима проверка решений задачи ЛП на здравый смысл.

2.5. Примеры применения моделей линейного программирования

2.5.1. Составление линейных моделей

Методы линейного программирования (ЛП) успешно применяются для решения широкого круга задач в промышленном производстве, социальных науках, на транспорте и т.д.

Пример. Сменно-суточное планирование работы автобусного парка (задача оптимального распределения ресурсов). Исследуются возможности более рациональной организации работы городского автобусного парка с целью снижения интенсивности внутригородского движения. На начальном этапе исследования было определено минимальное количество автобусов, которым можно удовлетворить существующую потребность в пассажирских перевозках.

Сбор и обработка необходимой информации позволили сделать вывод, что минимальное количество автобусов, которым можно удовлетворить потребность в перевозках, существенно меняется в течение суток. Требуемое количество автобусов можно считать величиной постоянной в пределах каждого из следующих друг за другом четырёхчасовых интервалов:

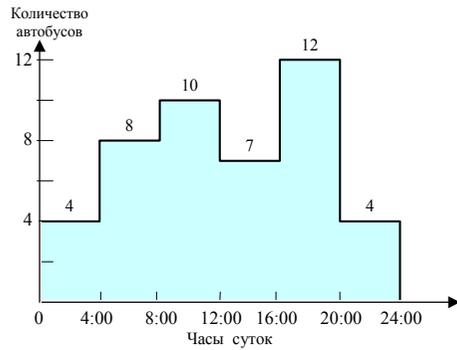
$$\begin{array}{lll} 0:01 - 4:00 - 4 & 8:01 - 12:00 - 10 & 16:01 - 20:00 - 12 \\ 4:01 - 8:00 - 8 & 12:01 - 16:00 - 7 & 20:01 - 24:00 - 4 \end{array}$$

В результате проведённого исследования было решено, что с учётом необходимых затрат времени на текущий ремонт и обслуживание непрерывное использование автобусов на линии должно продолжаться только по 8 часов в сутки.

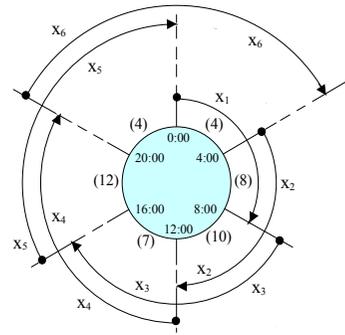
Решение. Построим гистограмму (столбцовую диаграмму) (рис. 2.6,а): Рассмотрим следующие случаи.

Случай I. Если ориентироваться на общепринятый трёхсменный график работы (8:01-16:00, 16:01-24:00, 0:01-8:00) и обозначить количество автобусов, выходящих на линию в I, II и III смены, через x_1, x_2, x_3 соответственно, то из гистограммы (рис. 2.6.а) можно увидеть, что $x_1 \geq 10, x_2 \geq 12, x_3 \geq 8$. По-

этому общее минимальное количество используемых автобусов $x_1 + x_2 + x_3 = 10 + 12 + 8 = 30$.



а) столбцовая диаграмма



б) круговая диаграмма

Рис. 2.6. Сменно-суточное планирование работы автобусного парка

Это решение приемлемо лишь для случая I. Но может оказаться, что выгоднее график работы, составленный на основе оптимального выбора начала каждой из смен.

Случай II. Примем, например, график работы, когда начало одной смены смещено относительно начала следующей смены на 4 часа. Такой график работы с перекрывающимися сменами показан на рис. 2.6,б для случая, когда смены начинаются в 0:01, 4:01, 8:01, 12:01, 16:01, 20:01, причём продолжительность смены составляет 8 часов. Теперь есть возможность идентифицировать переменные, для чего целесообразнее следующие обозначения для числа автобусов, выходящих на линию в разное время: x_1 – в 0:01, x_2 – в 4:01, x_3 – в 8:01, x_4 – в 12:01, x_5 – в 16:01, x_6 – в 20:01.

Соответствующая рис. 2.6,б математическая модель записывается следующим образом:

минимизировать $z = x_1 + x_2 + x_3 + x_4 + x_5 + x_6$
при ограничениях

$$\begin{aligned} x_1 + x_6 &\geq 4 \text{ (с 0:01 по 4:00);} & x_1 + x_2 &\geq 8 \text{ (с 4:01 по 8:00);} \\ x_2 + x_3 &\geq 10 \text{ (с 8:01 по 12:00);} & x_3 + x_4 &\geq 7 \text{ (с 12:01 по 16:00);} \\ x_4 + x_5 &\geq 12 \text{ (с 16:01 по 20:00);} & x_5 + x_6 &\geq 4 \text{ (с 20:01 по 24:00);} \\ x_j &\geq 0, \quad j = 1, 2, \dots, 6. \end{aligned}$$

Построенная модель приводит к следующему оптимальному решению: требуется только 26 автобусов, 10 из которых должны начинать работу в 4:01 (x_2), 12 – в 12:01 (x_4), 4 – в 20:01 (x_6), причём смены, начинающиеся в 0:01, 8:01, 16:01, исключаются (т. е. $x_1 = x_3 = x_5 = 0$). Таким образом, решение,

полученное в условиях возможности выбора начала смен, в отличие от решения, предполагающего использование традиционного трёхсменного графика, позволяет уменьшить суточную потребность в автобусах с 30 до 26.

2.5.2. Решение задачи линейного программирования симплекс-методом

Пример. На станции необходимо выгрузить маршрут однородного груза из 80 вагонов. Каждый из трёх грузовых фронтов может вместить определённое количество вагонов (табл. 2.1).

Таблица 2.1. Характеристика грузовых фронтов

| Грузовой фронт | Вместимость фронтов вагонов | Затрата локомотиво-часов на один вагон | Доход, ден. ед./ваг. |
|----------------|-----------------------------|--|----------------------|
| 1 | 35 | 0.2 | 3 |
| 2 | 40 | 0.4 | 5 |
| 3 | 25 | 0.3 | 4 |

Подаёт, расставляет, собирает и убирает их один локомотив, который работает 23 часа в сутки. Затраты локомотиво-часов маневровой работы, отнесённые на один вагон, различны для каждого грузового фронта. За выгрузку вагонов станция взимает с клиентов определённую плату. Но из-за различной технической оснащённости грузовых фронтов доход от выгрузки одного вагона не одинаков. Необходимо распределить вагоны по грузовым фронтам так, чтобы обеспечить максимальную выгрузку за сутки и максимальный доход.

Обозначим число вагонов, предназначенных для выгрузки на первом грузовом фронте, через x_1 , на втором – x_2 , на третьем – x_3 . Тогда целевая функция, выражающая общий доход станции от выгрузки вагонов

$$\max z = 3x_1 + 5x_2 + 4x_3$$

при ограничениях:

$x_1 + x_2 + x_3 \leq 80$ (сумма поданных вагонов не должна превышать их наличия);

$0,2x_1 + 0,4x_2 + 0,3x_3 \leq 23$ (общее время на обработку грузовых фронтов не должно превышать ресурсов локомотиво-часов);

$x_1 \leq 35; x_2 \leq 40, x_3 \leq 25$ (вместимость грузовых фронтов).

Математическая модель:

$$\begin{aligned} \max z &= 3x_1 + 5x_2 + 4x_3; \\ x_1 + x_2 + x_3 &\leq 80; \\ 0,2x_1 + 0,4x_2 + 0,3x_3 &\leq 23; \\ x_1 \leq 35; \quad x_2 \leq 40; \quad x_3 &\leq 25. \end{aligned}$$

Каноническая запись:

$$\begin{aligned} z - 3x_1 - 5x_2 - 4x_3 + 0y_1 + 0y_2 + 0y_3 + 0y_4 + 0y_5 &= 0; \\ x_1 + x_2 + x_3 + y_1 &= 80; \\ 0,2x_1 + 0,4x_2 + 0,3x_3 + y_2 &= 23; \\ x_1 + y_3 &= 35; \quad x_2 + y_4 = 40; \quad x_3 + y_5 = 25, \end{aligned}$$

где дополнительные (остаточные) переменные:

y_1 – число не поданных под выгрузку вагонов;

y_2 – количество неиспользованных локомотиво-часов;
 y_3, y_4, y_5 – неиспользованные вместимости грузовых фронтов.
 y_1, y_2, y_3, y_4, y_5 – базисные (опорные) переменные,
 x_1, x_2, x_3 – небазисные (свободные) переменные.

Точку $(0, 0, 0)$ используем как начальное допустимое решение.
 $y_1=80, y_2=23, y_3=35, y_4=40, y_5=25$ – базисный план;
 $x_1=0, x_2=0, x_3=0$ – начальный план.

I итерация – построение симплекс-таблицы:

| Базисные переменные | z | x_1 | x_2 | x_3 | y_1 | y_2 | y_3 | y_4 | y_5 | Решение | θ |
|---------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|-------------|
| z | 1 | -3 | -5↓ | -4 | 0 | 0 | 0 | 0 | 0 | 0 | |
| y_1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 80 | 80/1=80 |
| y_2 | 0 | 0,2 | 0,4 | 0,3 | 0 | 1 | 0 | 0 | 0 | 23 | 23/0,4=57,5 |
| y_3 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 35 | |
| y_4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 40 | 40/1=40 ← |
| y_5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 25 | |

Проверяем условие оптимальности: в z -строке среди коэффициентов при x_i $\max |x_i| = |x_2|$. Следовательно, столбец x_2 – ведущий, а x_2 – вводимая в базис переменная. Проверяем условие допустимости: $80/1=80$; $23/0,4=57,5$; $40/1=40$, $\min=40$, следовательно y_4 – ведущая строка, а y_4 – исключаемая из базиса переменная. На пересечении ведущих строки и столбца 1 – ведущий элемент.

II итерация – поиск нового базисного решения.

| Базисные переменные | z | x_1 | x_2 | x_3 | y_1 | y_2 | y_3 | y_4 | y_5 | Решение | θ |
|---------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|--------------|
| z | 1 | -3 | 0 | -4↓ | 0 | 0 | 0 | 5 | 0 | 200 | |
| y_1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | -1 | 0 | 40 | 40/1=40 |
| y_2 | 0 | 0,2 | 0 | 0,3 | 0 | 1 | 0 | -0,4 | 0 | 7 | 7/0,3=23,3 ← |
| y_3 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 35 | |
| x_2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 40 | |
| y_5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 25 | 25/1=25 |

III итерация – поиск нового базисного решения.

| Базисные переменные | z | x_1 | x_2 | x_3 | y_1 | y_2 | y_3 | y_4 | y_5 | Решение | θ |
|---------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|-----------------|
| z | 1 | -1/3↓ | 0 | 0 | 0 | 40/3 | 0 | -1/3 | 0 | 293,3 | |
| y_1 | 0 | 1/3 | 0 | 0 | 1 | -10/3 | 0 | 1/3 | 0 | 50/3 | (50/3)·3=50 |
| x_3 | 0 | 2/3 | 0 | 1 | 0 | 10/3 | 0 | -4/3 | 0 | 70/3 | (70/3)/(2/3)=35 |
| y_3 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 35 | 35/1=35 ← |
| x_2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 40 | |
| y_5 | 0 | -2/3 | 0 | 0 | 0 | -10/3 | 0 | 4/3 | 1 | 5/3 | |

IV итерация – поиск нового базисного решения.

| Базисные переменные | z | x_1 | x_2 | x_3 | y_1 | y_2 | y_3 | y_4 | y_5 | Решение | θ |
|---------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|----------------|
| z | 1 | 0 | 0 | 0 | 0 | 40/3 | 1/3 | -1/3↓ | 0 | 305 | |
| y_1 | 0 | 0 | 0 | 0 | 1 | -10/3 | -1/3 | 1/3 | 0 | 5 | 5/(1/3)=15 ← |
| x_3 | 0 | 0 | 0 | 1 | 0 | 10/3 | -2/3 | -4/3 | 0 | 0 | |
| x_1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 35 | |
| x_2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 40 | 40/1=40 |
| y_5 | 0 | 0 | 0 | 0 | 0 | -10/3 | 2/3 | 4/3 | 1 | 25 | 25/(4/3)=18,75 |

V итерация – поиск нового базисного решения.

| Базисные переменные | z | x_1 | x_2 | x_3 | y_1 | y_2 | y_3 | y_4 | y_5 | Решение | θ |
|---------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|---------|----------|
| z | 1 | 0 | 0 | 0 | 1 | 10 | 0 | 0 | 0 | 310 | |
| y_4 | 0 | 0 | 0 | 0 | 3 | -10 | -1 | 1 | 0 | 15 | |
| x_3 | 0 | 0 | 0 | 1 | | | | 0 | 0 | 20 | |
| x_1 | 0 | 1 | 0 | 0 | | | | 0 | 0 | 35 | |
| x_2 | 0 | 0 | 1 | 0 | | | | 0 | 0 | 25 | |
| y_5 | 0 | 0 | 0 | 0 | | | | 0 | 1 | 5 | |

Коэффициенты в z -строке неотрицательны. Следовательно, y_4, x_3, x_1, x_2, y_5 – оптимальный базисный план.

$$z^*_{max}=310; x^*_1=35, x^*_2=25, x^*_3=20.$$

2.5.3. Пример решения транспортной задачи методом потенциалов

Постановка задачи. Пусть имеется m станций отправления (поставщиков) и n станций назначения (потребителей) однородного или взаимозаменяемого груза. Заданы размеры отправления ($A_i, i = \overline{1, m}$) и прибытия ($B_j, j = \overline{1, n}$) по каждой из них, выраженные в вагонах, тоннах или других единицах измерения. Известны также стоимости перевозок единицы груза от каждой станции отправления до каждой станции назначения c_{ij} . Требуется составить такой план перевозок, чтобы общая стоимость транспортных расходов (z) была минимальной.

В линейном программировании под стоимостью понимают: расстояние, время, себестоимость перевозки. Решение, дающее минимум затрат по натуральным показателям, является также оптимальным по денежным затратам.

Задачу рассмотрим на следующем примере.

Оптимальное распределение порожних вагонов на полигоне сети с помощью методов линейного программирования.

Исходные данные:

1. Схема отделения железной дороги с указанием расстояния между стан-

станциями в километрах дана на рис. 2.7.

Станции А, Ж, Г – стыковые с соседними отделениями. (По станции Ж отделение принимает порожние вагоны, а по станции А – сдаёт.)

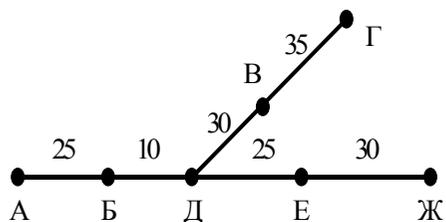


Рис. 2.7. Схема отделения железной дороги

2. Размеры выгрузки и погрузки крытых вагонов по станциям отделения:

а) выгрузка: А – 170; Б – 35; В – 80; Г – 25; Д – 150; Е – 80; Ж – 100;

б) погрузка: А – 290; Б – 110; В – 45; Г – 95; Д – 260; Е – 30.

3. По станции Ж отделение принимает в среднем в сутки 310 порожних вагонов, по станции А сдаёт 120 вагонов в сутки.

4. В качестве критерия «стоимости» при составлении оптимального плана распределения вагонов под погрузку принять кратчайшее расстояние (в километрах) между станциями избытка и недостатка вагонов.

Требуется:

1. Определить пункты избытка и недостатка порожних вагонов.

2. Установить тип модели транспортной задачи.

3. Составить матрицу кратчайших расстояний от станций избытка до станций недостатка порожних вагонов. Записать математическую модель.

4. Построить начальный план перевозок методом северо-западного угла и определить его общую стоимость.

5. Методом потенциалов найти оптимальный план распределения порожних вагонов на полигоне сети.

Ниже представлен алгоритм решения транспортной задачи в матричной форме.

Э т а п 1. Составим следующую таблицу (табл. 2.2):

Таблица 2.2. Распределение потребностей вагонов по грузовым фронтам

| Станция | Количество вагонов | | | | | | Станции избытка | Станции недостатка |
|---------|--------------------|-------|----------|-------|---------|------------|-----------------|--------------------|
| | выгрузка | приём | погрузка | сдача | избыток | недостаток | | |
| А | 170 | | 290 | 120 | | 240 | | 1 |
| Б | 35 | | 110 | | | 75 | | 2 |
| В | 80 | | 45 | | 35 | | 1 | |
| Г | 25 | | 95 | | | 70 | | 3 |
| Д | 150 | | 260 | | | 110 | | 4 |
| Е | 80 | | 30 | | 50 | | 2 | |
| Ж | 100 | 310 | | | 410 | | 3 | |
| Итого | 640 | 310 | 830 | 120 | 495 | 495 | 3 | 4 |

Из итоговой таблицы видно, что на трёх станциях отделения (В, Е, Ж) имеется избыток порожних вагонов, а на четырёх станциях (А, Б, Г, Д) их не хватает.

Э т а п 2. Суммарный избыток вагонов равен суммарному недостатку, что составляет 495 вагонов (т. е. выполняется условие общего баланса). Следовательно, решается задача закрытого типа или замкнутая транспортная модель (а значит, имеет решение) с матрицей размера 3×4.

Э т а п 3. Составим матрицу стоимостей (кратчайших расстояний между соответствующими станциями) перевозок от станций избытка до станций недостатка:

| | А | Б | Г | Д |
|---|----|----|-----|----|
| В | 65 | 40 | 35 | 30 |
| Е | 60 | 35 | 90 | 25 |
| Ж | 90 | 65 | 120 | 55 |

Математическая модель данной транспортной задачи формулируется следующим образом.

$$\text{Найти план перевозок } X = \|x_{ij}\| = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \end{pmatrix}$$

при условиях $x_{ij} \geq 0, i = 1, \dots, 3, j = 1, \dots, 4$ и ограничениях на избыток:

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &= 35; \\ x_{21} + x_{22} + x_{23} + x_{24} &= 50; \\ x_{31} + x_{32} + x_{33} + x_{34} &= 410, \end{aligned}$$

на недостаток:

$$\begin{aligned} x_{11} + x_{21} + x_{31} &= 240; & x_{12} + x_{22} + x_{32} &= 75; \\ x_{13} + x_{23} + x_{33} &= 70; & x_{14} + x_{24} + x_{34} &= 110 \end{aligned}$$

так, чтобы стоимость перевозок была минимальной:

$$\begin{aligned} \min z &= 65x_{11} + 40x_{12} + 35x_{13} + 30x_{14} + 60x_{21} + \\ &+ 35x_{22} + 90x_{23} + 25x_{24} + 90x_{31} + 65x_{32} + 120x_{33} + 55x_{34}. \end{aligned}$$

Э т а п 4. Построение начального базисного плана методом северо-западного угла (диагонального метода) (возможны и другие методы: метод наименьшей стоимости, метод двойного предпочтения и др.).

Заметим, что базисный план предполагает, что количество перевозок x_{ij} равно $m+n-1$. Начальным планом может быть любой план. Однако чем лучше он выбран, тем меньшее число итераций необходимо выполнить.

Назначение корреспонденций начинают с верхней левой клетки (северо-

западного угла) матрицы.

В верхнюю левую клетку записывается наименьшее из значений A_1 и B_1 ($\min(A_1, B_1)$). В нашем примере это 35. Остальные клетки будут с нулевыми перевозками (табл. 2.3).

Таблица 2.3. Построение начального плана перевозок

| | А | Б | Г | Д | A_i | |
|-------|------------------|-----|-----|------|-------|------------------------------|
| В | 35 | 65 | 40 | 35 | 30 | 35К |
| Е | 50 | 60 | 35 | 90 | 25 | 50К |
| Ж | 155 | 90 | 75 | 120 | 55 | 410 – 255 – 180 – 110 – К |
| B_j | 240, 205 155К | 75К | 70К | 110К | | |

Затраты при данном плане перевозок составят

$$z = 65 \cdot 35 + 60 \cdot 50 + 90 \cdot 155 + 65 \cdot 75 + 120 \cdot 70 + 55 \cdot 110 = 38550 \text{ ед.}$$

Число назначенных перевозок равно 6, это соответствует базисности начального плана.

Замечание. На практике часто встречаются задачи, когда надо исключать из рассмотрения и строку, и столбец до назначения последней перевозки (случай вырождения). Для сохранения базисности плана назначают нулевые перевозки. Исключают строку или столбец, в которых находятся клетки с большими стоимостями и в которые может попасть нулевая перевозка.

Э т а п 5. Построение оптимального плана методом потенциалов.

Полученный начальный план перевозок путём постепенного его улучшения доводится до оптимального.

Для уменьшения общей стоимости перевозок z по первоначальному плану необходимо переместить часть перевозок в клетки с меньшей стоимостью.

Потенциалами называется система чисел, приписанных по определённым правилам соответственно каждой строке i (u_i) и каждому столбцу j (v_j).

Шаг 1. Значения потенциалов определяются по стоимости перевозок в занятых клетках из условия оптимальности Канторовича:

$$\begin{cases} v_j - u_i = c_{ij}, & \text{если } x_{ij} > 0, \\ v_j - u_i \leq c_{ij}, & \text{если } x_{ij} = 0. \end{cases}$$

Если известен u_i , то $v_j = c_{ij} + u_i$, если известен v_j , то $u_i = v_j - c_{ij}$. Для строки с наибольшей стоимостью перевозок $c_{33} = 120$ положим $u_3 = 0$. Тогда будут вычислены и остальные потенциалы (табл. 2.4).

Шаг 2. Потенциалы вычислены. Затем делается проверка условия оптимальности ($v_j - u_i \leq c_{ij}$) для всех незагруженных клеток. Если в клетке условие оптимальности выполняется, то в этой клетке пометок не делается. В

клетках, в которых условие оптимальности не выполняется, вычисляется величина превышения, равная $\Delta_{ij} = v_j - u_i - c_{ij} > 0$.

Для клетки (1, 3) условие оптимальности не выполняется:

$$\Delta_{1,3} = v_3 - u_1 - c_{13} = 120 - 25 - 35 = 60 > 0.$$

Клетки с превышением стоимости называются потенциальными.

Шаг 3. Выбирается из потенциальных клетка с максимальным превышением. Строим замкнутый контур, начиная перемещаться в горизонтальном направлении с поворотами в загруженных клетках (1, 1), (3, 1), (3, 3) с таким расчётом, чтобы вернуться в исходную клетку. Пронумеруем клетки, начиная от клетки (1, 3). В вершинах контура с чётными номерами выбираем минимальную корреспонденцию, равную 35 в клетке (1, 1). На 35 увеличим корреспонденции в нечётных вершинах и уменьшим в чётных.

Новый (улучшенный) план представлен в табл. 2.5.

Таблица 2.4. Таблица потенциалов

| | $v_1=90$ | $v_2=65$ | $v_3=120$ | $v_4=55$ | |
|----------|----------|----------|-----------|----------|----|
| $u_1=25$ | 35 | 65 | 40 | 35 | 30 |
| $u_2=30$ | 50 | 60 | 35 | 90 | 25 |
| $u_3=0$ | 155 | 90 | 65 | 120 | 55 |
| | 155 | 75 | 70 | 110 | |

Таблица 2.5. Решение задачи

| | А | Б | Г | Д |
|---|-----|----|-----|-----|
| В | 25 | 40 | 35 | 30 |
| | | | 35 | |
| Е | 60 | 35 | 90 | 25 |
| | 50 | | | |
| Ж | 90 | 65 | 120 | 55 |
| | 190 | 75 | 35 | 110 |

Затраты при данном плане перевозок уменьшатся и составят

$$z_1 = z_0 - 60 \cdot 35 = 38550 - 60 \cdot 35 = 36450 \text{ ед.}$$

($60 \cdot 35$ – произведение превышения в потенциальной клетке на размеры назначенной в эту клетку перевозки).

Для небазисных клеток табл. 2.5 проверяем условие оптимальности. Проверка показала, что небазисных клеток план больше не содержит и является оптимальным с минимальной стоимостью перевозок.

Таким образом, для того чтобы затраты на перевозки порожних вагонов между станциями были минимальными, рекомендуется придерживаться полученного оптимального плана. В этом случае транспортные издержки составят 36450 ед.

Если матрица содержит не одну, а несколько потенциальных клеток, то шаги 2 и 3 повторяются до тех пор, пока во всех клетках не будет выполнено условие оптимальности. Отметим, что одна и та же транспортная задача может иметь несколько оптимальных планов, но суммарная стоимость перевозок по каждому такому плану будет одинаковой.

Другие методы: модифицированный распределительный метод, методы условных стоимостей, метод разрешаемых слагаемых и др.

Если имеем дело с задачей открытого типа, то её необходимо привести к задаче закрытого типа. Для этого вводится фиктивная станция назначения или отправления.

Помимо матричных способов существуют ещё сетевые способы решения транспортной задачи, которые позволяют учитывать наличную пропускную способность транспортной сети.

2.5.4. Пример графического решения задач линейного программирования

Графическим методом можно решать ЗЛП с $n > 2$ переменными, если в её канонической записи число неизвестных n и число линейно независимых уравнений m связаны соотношением $n - m \leq 2$. В этом случае каноническую форму задачи преобразовывают в симметричную, которая будет содержать не более двух переменных. Решая эту задачу графически, находят два компонента оптимального плана. Подставляя их в ограничения задач, определяют и остальные компоненты.

Пример. Двум погрузчикам разной мощности за 24 часа нужно погрузить на первой площадке 230 т, на второй – 168 т. Первый погрузчик на первой площадке может погрузить 10 т в час, на второй – 12 т. Второй – на каждой площадке может погрузить по 13 т в час. Стоимость работ, связанных с погрузкой 1 т, первым погрузчиком на первой площадке – 8 ден. ед., на второй – 7 ден. ед., вторым погрузчиком на первой площадке – 12 ден. ед., на второй – 13 ден. ед. Нужно составить план работы, т.е. найти, какой объём работ должен выполнить каждый погрузчик на каждой площадке, чтобы стоимость всех работ по погрузке была минимальной.

Решение. Обозначим через x_{ij} объём работ (в тоннах) i -го погрузчика ($i=1, 2$) на j -й площадке ($j=1, 2$). Условия задачи занесём в табл. 2.6.

Таблица 2.6. Исходные данные для задачи

| $i \backslash j$ | Π_1 | | Π_2 | | Лимит рабочего времени |
|------------------|----------|----------|----------|----------|------------------------|
| I погрузчик | 10 | 8 | 12 | 7 | |
| | x_{11} | x_{12} | x_{21} | x_{22} | |
| II погрузчик | 13 | 12 | 13 | 13 | 24 |
| | x_{21} | x_{22} | x_{21} | x_{22} | |
| Задание | 230 | | 168 | | |

Построим математическую модель задачи. Целевая функция описывает затраты, связанные с выполнением всех работ:

$$\min Z = 8x_{11} + 7x_{12} + 12x_{21} + 13x_{22}.$$

$$\text{Ограничения на лимиты рабочего времени: } \left. \begin{aligned} x_{11}/10 + x_{12}/12 &\leq 24, \\ x_{21}/13 + x_{22}/13 &\leq 24, \end{aligned} \right\}$$

$$\text{на необходимость выполнить задание: } \left. \begin{aligned} x_{11} + x_{12} &= 230, \\ x_{21} + x_{22} &= 168, \end{aligned} \right\}$$

условие неотрицательности: $x_{ij} \geq 0$ ($i, j=1, 2$).

Исключаем из модели переменные x_{21} и x_{22} . Из ограничений-равенств имеем

$$x_{21} = 230 - x_{11}, \quad x_{22} = 168 - x_{12}.$$

Подставив выражения для x_{21} и x_{22} в ограничения–неравенства и целевую функцию, получим ЗЛП с двумя переменными x_{11} и x_{12} :

$$\min Z = 4944 - 4x_{11} - 6x_{12};$$

$$\left. \begin{aligned} 6x_{11} + 5x_{12} &\leq 1440, \\ x_{11} + x_{12} &\geq 86, \\ x_{11} &\leq 230, \quad x_{12} \leq 168, \\ x_{11} &\geq 0, \quad x_{12} \geq 0. \end{aligned} \right\}$$

Очевидно, что целевая функция $Z = 4944 - 4x_{11} - 6x_{12}$ достигает минимального значения при условии, что $Z' = 4x_{11} + 6x_{12}$ принимает максимальное значение. Имеем задачу:

$$\left. \begin{aligned} \max Z' &= 4x_{11} + 6x_{12}; \\ 6x_{11} + 5x_{12} &\leq 1440, \\ x_{11} + x_{12} &\geq 86, \\ 0 &\leq x_{11} \leq 230, \\ 0 &\leq x_{12} \leq 168. \end{aligned} \right\}$$

Её графическое решение представлено на рис. 2.8.

Функция Z' достигает наибольшего значения при $x_{11}^* = 100, x_{12}^* = 168$.

Из выражений для x_{21} и x_{22} получим: $x_{21}^* = 130, x_{22}^* = 0$.

Итак, по оптимальному плану первый погрузчик должен погрузить 100 т на первой площадке и 168 т – на второй, второму погрузчику надлежит погрузить 130 т на первой площадке. Стоимость всех работ соста-

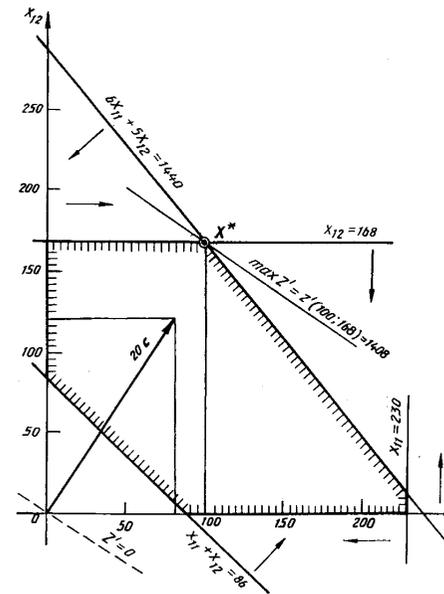


Рис. 2.8. Графическое решение задачи

вит 3536 ден. ед. ($Z^* = 4944 - 4 \cdot 100 - 6 \cdot 168 = 3536$).

2.5.5. Примеры анализа линейных моделей на чувствительность

Пример 1. Анализ предприятия по производству красок.

Содержательная формулировка задачи. Предприятие выпускает два вида a и b красок из сырья A и B . Всего имеется в наличии 6 тонн сырья A и 8 тонн сырья B . Спрос на краску типа a составляет на 1 тонну больше, чем на краску b , а спрос на краску типа b – не более двух тонн. Прибыль с единицы продукции красок a и b составляет соответственно $c_1 = 3$ руб. и $c_2 = 2$ руб. Необходимо найти объемы производства красок $x_1 \geq 0$ и $x_2 \geq 0$, которые максимизируют общую прибыль предприятия z .

Формальное описание. Поскольку задача проста, то этап составления таблицы опускаем и переходим к записи канонической формы:

$$\begin{aligned} \max z &= 3x_1 + 2x_2 \\ \text{при } x_1 + x_2 &\leq 6 && \text{(ограничение на сырье } A); \\ 2x_1 + x_2 &\leq 8 && \text{(ограничение на сырье } B); \\ -x_1 + x_2 &\leq 1 && \text{(ограничение спроса на краску } a); \\ x_2 &\leq 2 && \text{(ограничение спроса на краску } b); \\ x_1, x_2 &\geq 0 && \text{(ограничения на объемы производства красок).} \end{aligned} \quad (2.42)$$

Получив оптимальное решение для исходных условий, руководство предприятия интересуется, изменится ли оно. И если да, то какое будет новое оптимальное решение в следующих ситуациях:

- изменяются запасы сырья A и B ;
- изменяется спрос на краски a и b ;
- изменяются нормы расхода сырья A и B на тонну краски a и b ;
- возможно налажен выпуск краски еще одного вида.

Каждая из перечисленных ситуаций может повлиять на предыдущее оптимальное решение. Оно может быть: а) недопустимым; б) неоптимальным; в) недопустимым и неоптимальным.

Анализ влияния перечисленных ситуаций проведем, базируясь на вычислениях, использующих понятие двойственности задачи линейного программирования согласно такой последовательности действий:

Шаг 1. Решить исходную задачу ЛП табличным симплекс-методом.

Шаг 2. Применительно к рассматриваемым изменениям модели найти новые значения элементов симплекс-таблицы, используя соотношения двойственности.

Шаг 3. Если решение, представленное в новой симплекс-таблице, неоптимальное, перейти к шагу 4. Если это решение недопустимое, то переходим к шагу 5. Если полученное решение оптимальное, то процесс вычисления

закончен.

Шаг 4. С помощью симплекс-метода получить новое допустимое решение или показать, что оно неограниченное. Закончить вычисления.

Шаг 5. С помощью двойственного симплекс-метода получить допустимое решение или показать, что такого не имеется. Закончить вычисления.

Итак, запишем двойственную формулировку данного примера задачи ЛП:

$$\begin{aligned} z &= 6y_1 + 8y_2 + y_3 + 2y_4 \rightarrow \min; \\ y_1 - 2y_2 - y_3 &\geq 3; \quad 2y_1 + y_2 + y_3 + y_4 \geq 2; \\ y_1, y_2, y_3, y_4 &\geq 0. \end{aligned} \quad (2.43)$$

Составляем оптимальную симплекс-таблицу (табл. 2.7).

Таблица 2.7. Оптимальная симплекс-таблица прямой задачи

| C_j | — | — | 3 | 2 | 0 | 0 | 0 | 0 |
|-------|------------|-------|-------|-------|-------|-------|-------|-------|
| | B_x | A_0 | A_1 | A_2 | A_3 | A_4 | A_5 | A_6 |
| 2 | x_2 | 4/3 | 0 | 1 | 2/3 | -1/3 | 0 | 0 |
| 3 | x_1 | 10/3 | 1 | 0 | -1/3 | 2/3 | 0 | 0 |
| 0 | x_5 | 3 | 0 | 0 | -1 | 1 | 1 | 0 |
| 0 | x_6 | 2/3 | 0 | 0 | -2/3 | 1/3 | 0 | 1 |
| | Δ_i | 38/3 | 0 | 0 | 1/3 | 4/3 | 0 | 0 |

Векторы $A_4 - A_6$ соответствуют матрице обратной базисной на текущей итерации – A_x^{-1} .

Оптимальное значение переменных двойственной задачи определяем из табл. 2.7: $y_1 = \Delta_3 = 1/3$; $y_2 = \Delta_4 = 4/3$; $y_3 = \Delta_5 = 0$; $y_4 = \Delta_6 = 0$.

К недопустимости решения приводят два случая:

1. Изменение запасов ресурсов.
2. Введение новых ограничений.

Рассмотрим оба этих случая.

Случай 1. Анализируя найденное оптимальное решение прямой двойственной задачи, видим, что увеличение количества ресурсов сырья обоих видов выгодно. Особенно рост второго ресурса приводит к большему увеличению значений целевой функции:

$$z'_{\max} = z_{\max} + \Delta z; \quad \Delta z = y_i \Delta b_i.$$

Так, например, увеличение ресурса A на 1 единицу дает прирост целевой функции:

$$\Delta z_1 = \frac{1}{3} \cdot 1 = \frac{1}{3}; \quad z'_{\max} = z_{\max} + \Delta z_1 = 38/3 + 1/3 = 13.$$

Похожее увеличение ресурса B на 1 единицу даст еще больший прирост целевой функции:

$$\Delta z_2 = \frac{4}{3} \cdot 1 = \frac{4}{3}; \quad z'_{\max} = \frac{38}{3} + \frac{4}{3} = 14.$$

Таким образом, если имеется возможность, необходимо увеличить ресурсы. Например, увеличили второй вид ресурсов до 9. Новый вектор $A_0 = (6, 9, 1, 2)^T$. Новые значения переменных для нового условия будут равны:

$$\begin{pmatrix} x_2 \\ x_1 \\ x_5 \\ x_6 \end{pmatrix} = A_x^{-1} A_{0\text{нов}} = \begin{pmatrix} -2/3 & -1/3 & 0 & 0 \\ -1/3 & 2/3 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -2/3 & 1/3 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 6 \\ 9 \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 \\ 4 \\ 4 \\ 1 \end{pmatrix}$$

Решение осталось допустимым. При этом $z_{\max} = 14$.

Пример 2. Ситуация изменилась так, что ресурс A мы увеличим до 7, а ресурс B мы вынуждены снизить до 4. Ставится вопрос: каким будет новое решение и будет ли оно при этом допустимым. Новый вектор $A_0 = (7, 4, 1, 2)$. Новые значения переменных для нового условия будут равны:

$$\begin{pmatrix} x_2 \\ x_1 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 2/3 & -1/3 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -1/3 & 1/3 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 7 \\ 4 \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 10/3 \\ 1/3 \\ -2 \\ -4/3 \end{pmatrix}$$

Решение стало допустимым, так как $x_5 = -2$; $x_6 = -4/3$. Переходим к шагу 5. Заполняем с учетом нового вектора симплекс таблицу (табл. 2.8).

Таблица 2.8. Симплекс-таблица допустимого решения

| C_j | — | — | 3 | 2 | 0 | 0 | 0 | 0 |
|-------|----------|-------|-------|-------|-------|-------|-------|-------|
| | B_x | A_0 | A_1 | A_2 | A_3 | A_4 | A_5 | A_6 |
| 2 | x_2 | 10/3 | 1 | 0 | 2/3 | -1/3 | 0 | 0 |
| 3 | x_1 | 1/3 | 0 | 1 | -1/3 | 2/3 | 0 | 0 |
| 0 | x_5 | -2 | 0 | 0 | -1 | 1 | 1 | 0 |
| 0 | x_6 | -4/3 | 0 | 0 | -2/3 | 1/3 | 0 | 1 |
| | Δ | 23/3 | 0 | 0 | 1/3 | 4/3 | 0 | 0 |

За одну итерацию мы получаем новое оптимальное решение

$$x_1 = 1; \quad x_2 = 2; \quad z_{\max} = 7.$$

Случай 2. Добавление нового ограничения может привести к двум ситуациям: а) новое ограничение избыточное, поэтому решение не изменится; б) новое ограничение связывающее. Пусть новым ограничением будет $x_1 \leq 4$. Очевидно, что новое найденное решение $x_1 = 10/3$, это ограничение выполня-

ется и оно является уже избыточным. Допустим, $x_1 \leq 3$. Тогда при оптимальном решении $x_1 = 10/3$ оно не будет выполняться. Поэтому необходимо с учетом $x_1 \leq 3$, $x_1 + x_3 = 3$; $x_7 \geq 0$ найти новое оптимальное решение. Поскольку x_1 в найденном оптимальном решении является базисной переменной, то выразим ее во вводимом ограничении через небазисные переменные. Из оптимальной симплекс-таблицы 2.7 имеем следующее равенство:

$$x_1 - \frac{1}{3}x_3 + \frac{2}{3}x_4 = \frac{10}{3}.$$

$$\text{Тогда } \frac{10}{3} + \frac{1}{3}x_3 - \frac{2}{3}x_4 + x_7 = 3 \text{ или } \frac{1}{3}x_3 - \frac{2}{3}x_4 + x_7 = -\frac{1}{3}.$$

Введем это дополнительное ограничение в симплекс-таблицу 2.7, приняв в качестве еще одной базисной переменной x_7 . Выполняем шаг 5 и за одну итерацию получаем новое оптимальное решение.

К нарушению оптимальности уже найденного решения могут привести:

- 1) изменение коэффициентов целевой функции;
- 2) изменение удельных расходов ресурсов;
- 3) добавление нового вида продукции.

Исследуем каждый из этих трех случаев.

Случай 1. Изменяются коэффициенты целевой функции, соответствующие как базисным переменным, так и небазисным. Проверим, останется ли план оптимальным, если изменить коэффициенты при базисных переменных. Пусть новая целевая функция будет $z = 6x_1 + 4x_2$. Найдем значение переменных двойственной задачи по формуле

$$Y^{\text{опт}} = C^{\text{опт}} A_x^{-1},$$

где A_x^{-1} – матрица, обратная матрице, составленной из компонентов векторов, входящих в последний базис, при котором получен оптимальный план. Используя эти значения, определим новые оценки Δ_j :

$$Y^{\text{опт}} = (4, 6, 0, 0) \cdot \begin{pmatrix} 2/3 & -1/3 & 0 & 0 \\ 1/3 & 2/3 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -1/3 & 1/3 & 0 & 1 \end{pmatrix} = (2/3, 8/3, 0, 0),$$

т. е. $y_1 = 2/3$; $y_2 = 8/3$; $y_3 = 0$; $y_4 = 0$. Проверим оценки Δ_j . Имеем:

- для Δ_1 : $y_1 + 2y_2 - y_3 \geq 6$; $\Delta_1 = 2/3 - 2 \cdot 8/3 - 0 - 6 = 0$;
- для Δ_2 : $2y_1 + y_2 + y_3 + y_4 \geq 4$; $\Delta_2 = 2 \cdot 2/3 + 8/3 + 0 + 0 - 4 = 0$;
- для Δ_3 : $y_1 \geq 0$; $\Delta_3 = 2/3 - 0 = 2/3$;
- для Δ_4 : $y_2 \geq 0$; $\Delta_4 = 8/3 - 0 = 8/3$;
- для Δ_5 : $y_3 \geq 0$; $\Delta_5 = 0 - 0 = 0$;

- для $\Delta_6: y_4 \geq 0; \Delta_6 = 0 - 0 = 0$.

Все значения оценок положительны. Поэтому план $(x_1 = -10/3; x_2 = 4/3)$ остался оптимальным. Изменилась только величина $z_{\max} = 6x_1 + 4x_2 = 24 \frac{2}{3}$.

Отметим, что необходимо вычислять только оценки Δ_j для небазисных переменных, поскольку значения оценок для базисных переменных всегда остаются равными нулю.

Рассмотрим ситуацию, приводящую к нарушению признака оптимальности. Пусть целевая функция станет равной $z = 4x_1 + x_2$.

Тогда

$$y^{\text{опт}} = (1, 4, 0, 0) \cdot \begin{bmatrix} 2/3 & -1/3 & 0 & 0 \\ -1/3 & 2/3 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -2/3 & 1/3 & 0 & 1 \end{bmatrix} = (-2/3, 7/3, 0, 0),$$

т.е. $y_1 = -2/3; y_2 = 7/3; y_3 = 0; y_4 = 0$.

Определим оценки Δ_j . Для первых двух базисных переменных x_1 и x_2 оценки равны $\Delta_1 = 0; \Delta_2 = 0$. Далее имеем:

$$y_1 = \Delta_3; y_2 = \Delta_4; y_3 = \Delta_5; y_4 = \Delta_6; \\ \Delta_3 = -2/3; \Delta_4 = 7/3; \Delta_5 = 0; \Delta_6 = 0.$$

Следовательно, план, представленный табл. 2.7, с новыми коэффициентами целевой функции стал неоптимальным. Выполняем шаг 4. Заполним новую симплекс-таблицу, изменив в табл. 2.7 значения индексной строки Δ_j и $C_1=4; C_2=1$. На следующей итерации получаем оптимальный план (табл. 2.9).

Таблица 2.9. Оптимальный план задачи

| C_j | — | — | 4 | 1 | 0 | 0 | 0 | 0 |
|-------|----------|-------|-------|-------|-------|-------|-------|-------|
| | B_x | A_0 | A_1 | A_2 | A_3 | A_4 | A_5 | A_6 |
| 0 | x_3 | 2 | 0 | 3/2 | 1 | -1/2 | 0 | 0 |
| 4 | x_1 | 4 | 1 | 1/2 | 0 | 1/2 | 0 | 0 |
| 0 | x_5 | 5 | 0 | 3/2 | 0 | 1/2 | 1 | 0 |
| 0 | x_6 | 2 | 0 | 1 | 0 | 0 | 0 | 1 |
| | Δ | 16 | 0 | 1 | 0 | 2 | 0 | 0 |

Исследование случаев изменения значений коэффициентов целевой функции при небазисных переменных выполняется с помощью аналогичных вычислений.

Случай 2. Изучение влияния изменения удельных расходов ресурсов на оптимальное решение ограничим коэффициентами и представим следующим образом: найти $\max z = 4x_1 + x_2$ при ограничениях (2.42). Оптималь-

ное решение функции представлено табл. 2.9. Изменим потребление ресурсов A и B для производства тонны краски вида B . Изменим коэффициенты при x_2 с 2 и 1 на коэффициенты 5 и 3 соответственно. Ограничение двойственной задачи, соответствующее x_2 , примет вид: $5y_1 + 3y_2 + y_3 + y_4 \geq 1$.

Определяем из табл. 2.9 значения y_1, y_2, y_3, y_4 и затем новую оценку Δ_2 :

$$y_1 = \Delta_3 = 0; y_2 = \Delta_4 = 2; y_3 = \Delta_5 = 0; y_4 = \Delta_6 = 0; \\ \Delta_2 = 5y_1 + 3y_2 + y_3 + y_4 - 1 = 5.$$

Поскольку Δ_2 неотрицательно, то указанные изменения не нарушают оптимального решения табл. 2.9. Если бы оценка получилась отрицательной, то необходимо было бы выполнить шаг 4.

Случай 3. Добавление еще одного вида деятельности предприятия, выражающееся в выпуске краски третьего вида. Обозначим объем краски третьего вида через x_7 . Тогда задача ЛП запишется следующим образом:

$$\max z = 3x_1 + 2x_2 + 2x_7; \\ x_1 + 2x_2 + x_7 \leq 6; 2x_1 + x_2 + 3/4x_7 \leq 8; \\ -x_1 + x_2 - x_7 \leq 1; 0 \leq x_2 \leq 2; \\ x_1 \geq 0; x_7 \geq 0. \quad (2.44)$$

Добавление нового вида производственной деятельности эквивалентно одновременному изменению целевой функции и коэффициентов в ограничениях. Предположим, что переменная x_7 ранее присутствовала с коэффициентом, равным нулю, а в модифицированной модели 2.44 коэффициенты x_7 приняли соответствующие значения. Такой подход позволяет считать переменную x_7 для оптимального решения (см. табл. 2.7) небазисной. Найдем оценку Δ_7 и проверим оптимальность новой задачи. Из табл. 2.7 определяем:

$$y_1 = \Delta_3 = 1/3; y_2 = \Delta_4 = 4/3; y_3 = \Delta_6 = 0; \\ y_1 + 3/4y_2 - y_3 \geq 2; \Delta_7 = 1/3 + 3/4 \cdot 4/3 - 1 - 2 = -5/3.$$

Следовательно, введением в базис новой переменной x_7 план может быть улучшен. Приступаем к шагу 4. В симплекс-таблицу 2.7 добавляем столбец A_7 , ассоциированный с переменной x_7 , $\Delta_7 = -5/3; C_7 = 2$:

$$A_7 = \begin{bmatrix} 2/3 & -1/3 & 0 & 0 \\ -1/3 & 2/3 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ -2/3 & 1/3 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 3/4 \\ -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 5/12 \\ 1/6 \\ -3/4 \\ -5/2 \end{bmatrix}.$$

За одну итерацию получаем новое оптимальное решение: $x_1 = 14/5; x_2 = 0; x_7 = 10/5; z = 14 \frac{4}{5}$.

3. СЕТЕВЫЕ И ГРАФОВЫЕ МОДЕЛИ

3.1. Модели на графах

Объектом исследования является сеть, состоящая из узлов и линий связи. Предполагается, что в сети имеется два особых узла: вход в сеть (исток) и выход из сети (сток). В качестве аппарата формализации используется теория графов и специально разработанный алгоритм решения таких задач. Наиболее часто решаются следующие типы задач:

- нахождение максимального потока грузов через транспортную сеть;
- нахождение кратчайшего (или самого дешёвого) пути между произвольными узлами сети;
- построение самой дешёвой сети дорог, линий передач, трубопроводов;
- поиск узких мест в транспортной сети;
- оптимальное назначение исполнителей по работам.

Технология решения этих задач основывается на следующих понятиях и обозначениях. Каждая дуга графа, соединяющая два узла x_i и x_j , характеризуется длиной $l(x_i, x_j)$. Каждый из узлов сети (x_i) называют вершиной графа G , представляющего собой всю сеть. Дуги могут быть как ориентированными, обозначаемыми на графе G в виде стрелок, так и неориентированными. Общее число вершин графа G равно n , и индексы вершин x_i меняются от $i = 1$ до $i = n$.

Введём понятие **транспортная сеть** – связный ориентированный мультиграф без петель, в котором:

- существует одна и только одна вершина x_0 , называемая входом сети (исток);
- существует одна и только одна вершина x_n , называемая выходом сети (сток);
- каждой дуге $g_{ij}(x_i, x_j)$ отнесено целое число $C(x_i, x_j) \geq 0$, называемое пропускной способностью.

Для произвольной вершины x_i множество дуг, входящих в неё, обозначим через $U_{x_i}^-$, а множество дуг, выходящих из x_i , обозначим через $U_{x_i}^+$.

Тогда **поток** $\varphi(g_{ij})$ транспортной сети назовём **целочисленную** функцию, определённую на множестве дуг графа $G(x)$, которая удовлетворяет следующим свойствам:

- 1) $\varphi(g_{ij}) \geq 0$, $g \in \{x_i, G(x_i), i = \overline{1, n}\}$;
- 2) $\sum_{g_{ij} \in U_x^-} \varphi(g_{ij}) - \sum_{g_{ij} \in U_x^+} \varphi(g_{ij}) = 0$, для $x \neq x_0$, $x \neq x_n$;

$$3) \varphi(g_{ij}) \leq C(g), \quad g \in U.$$

Свойство 2 означает известное условие сохранения вещества в промежуточных узлах x_i . Оно показывает, что в любой промежуточной вершине транспортной сети не создаётся и не исчезает вещество. Свойство 3 показывает, что в единицу времени реальное количество вещества (грузов) не может быть больше пропускной способности дуги.

Приток вещества в вершине x_n представляет собой величину транспортного потока в сети и определяется по формуле

$$\varphi_{x_n} = \sum_{g_{ij} \in U_{x_n}^-} \varphi(g_{ij}) = \sum_{g_{ij} \in U_{x_0}^+} \varphi(g_{ij}) = 0.$$

Если сделать разрез графа G через множество вершин $\{x_i\} = A$, то множество дуг, заходящих в множество вершин A , назовём **разрезом сети**. **Пропускная способность разреза** равна сумме пропускных способностей дуг, входящих в множество A :

$$C(U_A^-) = \sum_{g \in U_A} C(g).$$

Любой элемент вещества (груз), движущийся от x_0 к x_n , пройдёт хотя бы один раз по некоторой дуге разреза U_A^- . Тогда каковы бы ни были поток $\varphi(g_{ij})$ и разрез сети U_A^- , всегда имеет место неравенство

$$\varphi_{x_n} \leq C(U_A^-).$$

Существует лемма [2], которую мы приводим без доказательства.

Если для какого-либо разреза U_A^- имеет место равенство $\varphi_{x_n} = C(U_A^-)$, то 1) поток имеет наибольшую величину; 2) разрез U_A^- обладает наименьшей пропускной способностью, равной сумме пропускных способностей дуг.

3.1.1. Задача о наибольшем потоке в сети

Пусть сеть представлена графом $G(x)$ с множеством рёбер U , n вершинами x_i , у которого x_0 и x_n – соответственно вход и выход сети. Для каждого ребра сети (x_i, x_j) или дуги задана пропускная способность

$$c_{ij} = \begin{cases} 0 & \text{при } (x_i, x_j) \notin U, \\ C(x_i, x_j), & \text{когда } (x_i, x_j) \in U. \end{cases}$$

Надо найти такую систему чисел $\varphi(x_i, x_j)$, представляющих фактические

значения потоков в дугах графа G , для которой справедливо неравенство

$$0 \leq \varphi(x_i, x_j) \leq c_{ij}.$$

Это неравенство означает тот очевидный факт, что поток по ребру (x_i, x_j) не может быть больше пропускной способности дуги. Исходя из третьего условия определения потока следует, что любой поток, входящий в x_i , равен потоку, выходящему из вершины x_i :

$$\sum_{k=0}^n \varphi(x_k, x_i) = \sum_{j=1}^n \varphi(x_i, x_j) = \varphi(U_{x_i}^-) = \varphi(U_{x_i}^+). \quad (3.1)$$

Условие максимальности потока на выходе транспортной сети будет иметь вид

$$\varphi_{\max} = \max \sum_{i=1}^n \varphi(x_i, x_n). \quad (3.2)$$

Это классическая задача ЛП, но обычно труднорешаема из-за плохой сходимости алгоритма симплекс-метода, требующая много итераций (жордановых исключений). Поэтому американские учёные Форд и Фалкерсон предложили более простой алгоритм её решения. Идея этого алгоритма состоит в постепенном увеличении потока до тех пор, пока он не станет максимальным. Единственное ограничение на применение алгоритма Форда-Фалкерсона состоит в том, что $C(x_i, x_j)$ должны быть целыми числами. Поэтому и величина потока представляет целочисленную функцию.

Введём ещё несколько определений. Дуга g называется *насыщенной*, если $\varphi(g)=C(g)$. Поток называется *полным*, если любой путь из x_0 в x_n содержит хотя бы одну насыщенную дугу. Если некоторый *начальный* поток является неполным, то можно найти такой путь S , который соединяет x_0 с x_n и не содержит насыщенной дуги.

Алгоритм основывается на теореме Форда-Фалкерсона: в любой транспортной сети максимальный поток равен минимальной пропускной способности. Если поток максимален, то найдётся такое сечение, пропускная способность которого равна мощности потока. Доказывается эта теорема применением алгоритма Форда-Фалкерсона. Согласно этому алгоритму, начиная с некоторого начального неполного потока, по итеративному алгоритму можно получить полный поток, если прибавлять к различным значениям потоков $l(g)$ пути S_i минимальное из чисел $[c(g)-l(g)]$, которые вычислены по этому пути S_i . После такой операции путь S_i уже будет содержать хотя бы одну насыщенную дугу. Если таким же образом поступить с другими путями S_j , то в конце концов получим полный поток. Поэтому алгоритм определения максимального потока состоит из следующих шагов:

1. Строим начальный поток X^0 .

2. Проверяем, попал ли узел x_n в множество узлов S , которые достижимы по ненасыщенным рёбрам из x_0 . Если узел x_n не попал, то считают, что построенный поток максимален, и алгоритм расчёта останавливается.

3. Если узел x_n попал в множество S , то выделяют путь S_j , состоящий из ненасыщенных рёбер и ведущий грузы из x_0 в x_n .

4. Увеличивают поток через каждое ребро этого пути на величину $\min(c_{ij} - \varphi_{ij})$.

5. Строят новый поток X и переходят к шагу 2.

Обычно сеть задаётся матрицей пропускных способностей $\Sigma = \|c_{ij}\|$ всех рёбер сети U . Задавшись $k=0$, затем вычисляют на k -м шаге матрицу значений потоков на дугах $X^k = \|\varphi_{ij}^k\|$. Строят матрицу разностей $\Sigma - X^k = \|c_{ij} - \varphi_{ij}^k\|$. В этой матрице насыщенным рёбрам при потоке X^k будут соответствовать нулевые элементы, а ненасыщенным – ненулевые элементы. Поэтому вычисление матрицы $\|\Sigma - X^k\|$ достаточно как для построения множества узлов, по которым вещество из x_0 достигает по ненасыщенным рёбрам до x_n , так и для построения последовательности ненасыщенных рёбер S_1 , позволяющих достигать x_n из x_0 .

Технология составления этих списков следующая:

– сначала составляют список узлов, в каждый из которых ведёт ненасыщенное ребро из вершины i ;

– далее для каждого i -го узла составляют свой список узлов, в каждый из которых из i -го узла ведёт ненасыщенное ребро (за исключением тех узлов, которые уже вошли в ранее составленные списки), и так далее. Этот процесс выписывания списков заканчивается в двух случаях. Либо появился узел x_n , что означает продолжение работы алгоритма, либо в список выписанных узлов не попал узел x_n , что означает конец расчётов. Поэтому с вычислительной точки зрения алгоритм Форда-Фалкерсона реализует такую последовательность действий:

1. Вычисление матрицы $\|\Sigma - X^k\|$.

2. Построение списков вершин.

3. Нахождение последовательности ненасыщенных рёбер, соединяющих x_0 с x_n .

4. Нахождение величины дополнительного потока $\Delta\varphi = \min\{c_{ij} - \varphi_{ij}\}$.

5. Построение нового потока X^{k+1} .

6. Если в списке отсутствует x_n , то поток X_k – максимален.

Пример 1. Найти максимальный поток в сети, заданной графом G , который представлен на рис. 3.1. Задана матрица пропускных способностей $\Sigma = \|c_{ij}\|$, которая имеет вид

| Σj $i \downarrow \rightarrow$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | x_n |
|--|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 1 | 4 | 0 | 0 | 5 | 0 | 0 | 0 |
| x_1 | 0 | 0 | 0 | 1 | 1 | 0 | 2 | 0 | 0 |
| x_2 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 3 | 0 |
| x_3 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 |
| x_4 | 0 | 1 | 0 | 0 | 0 | 3 | 1 | 0 | 0 |
| x_5 | 4 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 2 |
| x_6 | 0 | 2 | 0 | 3 | 1 | 0 | 0 | 2 | 6 |
| x_7 | 0 | 0 | 2 | 0 | 0 | 0 | 2 | 0 | 2 |
| x_n | 0 | 0 | 0 | 0 | 0 | 2 | 2 | 2 | 0 |

На рис. 3.1 указаны на рёбрах g_{ij} значения пропускных способностей в виде вектора (c_{ij}, c_{ji}) , компонентами которого являются пропускные способности соответственно в прямом и обратном направлениях.

Выберем в качестве начального потока X^0 распределение пяти единиц вещества движущихся по ненасыщенным дугам согласно матрице.

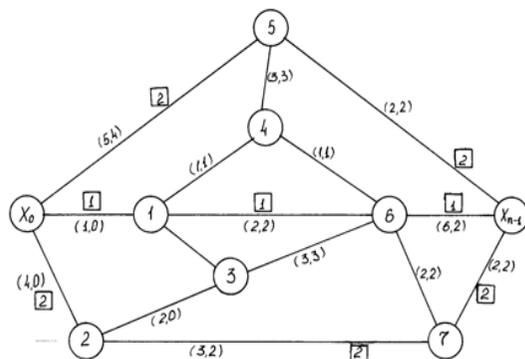


Рис. 3.1. Граф сети, заданной матрицей пропускных способностей дуг

| $X^0 j$ $i \downarrow \rightarrow$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | x_n | Итого |
|---------------------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 1 | 2 | 0 | 0 | 2 | 0 | 0 | 0 | 5 |
| x_1 | -1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x_2 | -2 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x_4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x_5 | -2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_6 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| x_7 | 0 | 0 | -2 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | -2 | -1 | -2 | 0 | -5 |

На рис. 3.1. значения начального потока показаны квадратными скобками. Уход вещества из узла x_i обозначаем со знаком "+", а приход вещества в узел x_j задаётся со знаком "-" в матрицах X^k .

Первый шаг алгоритма. Составим матрицу $\|\Sigma - X^0\|$:

| $\Sigma - X^0$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | x_n |
|----------------|-------|-------|----------|----------|-------|-------|----------|-------|----------|
| x_0 | 0 | 0 | <u>2</u> | 0 | 0 | 3 | 0 | 0 | 0 |
| x_1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| x_2 | 2 | 0 | 0 | <u>2</u> | 0 | 0 | 0 | 1 | 0 |
| x_3 | 0 | 0 | 0 | 0 | 0 | 0 | <u>3</u> | 0 | 0 |
| x_4 | 0 | 1 | 0 | 0 | 0 | 3 | 1 | 0 | 0 |
| x_5 | 6 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 |
| x_6 | 0 | 3 | 0 | 3 | 1 | 0 | 0 | 2 | <u>5</u> |
| x_7 | 0 | 0 | 4 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | 4 | 3 | 4 | 0 |

Составим списки узлов: $(x_0 \rightarrow x_2, x_5)$; $(x_2 \rightarrow x_3, x_7)$; $(x_5 \rightarrow x_4)$; $(x_3 \rightarrow x_6)$; $(x_7 \rightarrow ?)$; $(x_4 \rightarrow x_1)$; $(x_6 \rightarrow x_n)$.

Здесь "?" означает, что после x_7 нет продолжения в графе G . Поскольку узел x_n встретился, то алгоритм вычисления продолжается. Последовательность ненасыщенных рёбер имеет вид (x_0, x_2) ; (x_2, x_3) ; (x_3, x_6) ; (x_6, x_n) .

Приращение дополнительного потока равно: $\min \{2, 2, 3, 5\} = 2$. Добавим величину 2 в матрице X^0 в местах, где в матрице $\|\Sigma - X^0\|$ элементы выделены и подчёркнуты. Матрица нового потока X^1 будет иметь вид:

| X^1 | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | x_n | Итого |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 1 | 4 | 0 | 0 | 2 | 0 | 0 | 0 | 7 |
| x_1 | -1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x_2 | -4 | 0 | 0 | 2 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_3 | 0 | 0 | -2 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| x_4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| x_5 | -2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_6 | 0 | -1 | 0 | -2 | 0 | 0 | 0 | 0 | 3 | 0 |
| x_7 | 0 | 0 | -2 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | -2 | -3 | -2 | 0 | -7 |

Второй шаг алгоритма. Вычислим матрицу $\|\Sigma - X^1\|$:

| $\Sigma - X^1$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | x_n |
|----------------|-------|-------|-------|-------|----------|----------|----------|-------|----------|
| x_0 | 0 | 0 | 0 | 0 | 0 | <u>3</u> | 0 | 0 | 0 |
| x_1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| x_2 | 4 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| x_3 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 1 | 0 |
| x_4 | 0 | 1 | 0 | 0 | 0 | 3 | <u>1</u> | 0 | 0 |
| x_5 | 6 | 0 | 0 | 0 | <u>3</u> | 0 | 0 | 0 | 0 |
| x_6 | 0 | 3 | 0 | 5 | 1 | 0 | 0 | 2 | <u>3</u> |
| x_7 | 0 | 0 | 4 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | 4 | 4 | 5 | 0 |

Составим списки узлов: $(x_0 \rightarrow x_5)$; $(x_5 \rightarrow x_4)$; $(x_4 \rightarrow x_1, x_6)$; $(x_1 \rightarrow x_3)$; $(x_6 \rightarrow x_7, x_n)$. Поскольку узел x_n встретился в списке, алгоритм вычисления продолжается. Последовательность ненасыщенных рёбер имеет вид (x_0, x_5) ; (x_5, x_4) ; (x_4, x_6) ; (x_6, x_n) . Приращение дополнительного потока равно $\min \{3, 3, 1, 3\} = 1$. Добавим величину 1 в матрице X^1 соответственно в местах, где в матрице $\|\Sigma - X^1\|$ элементы выделены и подчёркнуты. Матрица нового потока имеет вид:

| X^2 | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | X_N | Итого |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 1 | 4 | 0 | 0 | 3 | 0 | 0 | 0 | 8 |
| x_1 | -1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x_2 | -4 | 0 | 0 | 2 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_3 | 0 | 0 | -2 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| x_4 | 0 | 0 | 0 | 0 | 0 | -1 | 1 | 0 | 0 | 0 |
| x_5 | -3 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 |
| x_6 | 0 | -1 | 0 | -2 | -1 | 0 | 0 | 0 | 4 | 0 |
| x_7 | 0 | 0 | -2 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | -2 | -4 | -2 | 0 | -8 |

Третий шаг алгоритма. Вычислим матрицу $\|\Sigma - X^2\|$:

| $\Sigma - X^2$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | X_N |
|----------------|-------|----------|-------|-------|----------|----------|----------|-------|----------|
| x_0 | 0 | 0 | 0 | 0 | 0 | <u>2</u> | 0 | 0 | 0 |
| x_1 | 1 | 0 | 0 | 1 | 1 | 0 | <u>1</u> | 0 | 0 |
| x_2 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| x_3 | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| x_4 | 0 | <u>1</u> | 0 | 0 | 0 | 4 | 0 | 0 | 0 |
| x_5 | 7 | 0 | 0 | 0 | <u>2</u> | 0 | 0 | 0 | 0 |
| x_6 | 0 | 3 | 0 | 5 | 2 | 0 | 0 | 2 | <u>2</u> |
| x_7 | 0 | 0 | 4 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | 4 | 6 | 4 | 0 |

Составим списки узлов: $(x_0 \rightarrow x_5)$; $(x_5 \rightarrow x_4)$; $(x_4 \rightarrow x_1)$; $(x_1 \rightarrow x_3, x_6)$; $(x_3 \rightarrow x_2)$; $(x_6 \rightarrow x_7, x_n)$. Поскольку узел x_n встретился в списке, алгоритм вычисления продолжается. Последовательность ненасыщенных рёбер имеет вид: (x_0, x_5) ; (x_5, x_4) ; (x_4, x_1) ; (x_1, x_6) ; (x_6, x_n) . Приращение дополнительного потока равно: $\min \{2, 2, 1, 1, 2\} = 1$. Добавим величину 1 в матрице X^2 соответственно в местах, где в матрице $\|\Sigma - X^2\|$ элементы выделены и подчёркнуты. Матрица нового потока имеет вид:

| X^3 | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | X_N | Итого |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 1 | 4 | 0 | 0 | 4 | 0 | 0 | 0 | 9 |
| x_1 | -1 | 0 | 0 | 0 | -1 | 0 | 2 | 0 | 0 | 0 |
| x_2 | -4 | 0 | 0 | 2 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_3 | 0 | 0 | -2 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| x_4 | 0 | 1 | 0 | 0 | 0 | -2 | 1 | 0 | 0 | 0 |
| x_5 | -4 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 2 | 0 |
| x_6 | 0 | -2 | 0 | -2 | -1 | 0 | 0 | 0 | 5 | 0 |
| x_7 | 0 | 0 | -2 | 0 | 0 | 0 | 0 | 0 | 2 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | -2 | -5 | -2 | 0 | -9 |

Четвёртый шаг алгоритма. Вычислим матрицу $\|\Sigma - X^3\|$:

| $\Sigma - X^3$ | x_0 | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | X_N |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| x_0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| x_1 | 1 | 0 | 0 | 1 | 2 | 0 | 0 | 0 | 0 |
| x_2 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| x_3 | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| x_4 | 0 | 0 | 0 | 0 | 0 | 5 | 0 | 0 | 0 |
| x_5 | 8 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| x_6 | 0 | 4 | 0 | 5 | 2 | 0 | 0 | 2 | 1 |
| x_7 | 0 | 0 | 4 | 0 | 0 | 0 | 2 | 0 | 0 |
| x_n | 0 | 0 | 0 | 0 | 0 | 4 | 7 | 4 | 0 |

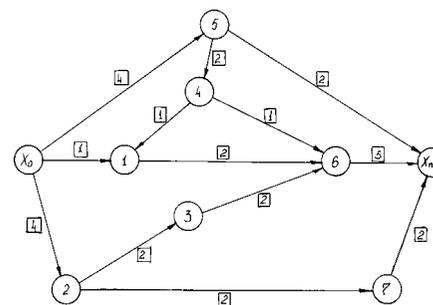


Рис. 3.2. Граф максимального потока

Составим списки узлов: $(x_0 \rightarrow x_5)$; $(x_5 \rightarrow x_4)$; $(x_4 \rightarrow ?)$. Поскольку узел x_n не встретился в списке, алгоритм вычисления максимального потока останавливается. Поток X^3 , равный 9 единицам, является максимальным. Окончательный вид максимального значения транспортного потока представлен на рис. 3.2 в виде ориентированного графа, на рёбрах которого указаны фактические значения потоков.

Как видно из рис. 3.2, ребро (x_1, x_3) является лишним, поскольку через него нет потока вещества. Это позволяет достичь экономии материалов за счёт его исключения.

3.1.2. Нахождение кратчайшего пути в транспортной сети

В зависимости от содержания задачи может быть два случая: когда рёбра графа G единичной длины; когда рёбра графа произвольной длины. Для ка-

ждого из этих случаев разработаны две разновидности алгоритма решения задачи. Рассмотрим первый случай. Транспортная сеть представлена графом G_1 , в котором длины рёбер одинаковы и равны некоторой постоянной величине ($l_i = \text{const}$). Для простоты изложения будем считать, что $l_1 = 1$. Каждому узлу x_i графа G_1 приписываем индекс λ_i , равный длине кратчайшего пути из x_i в конечный узел x_n .

Алгоритм основан на последовательном приписывании индексов, проходя в обратном направлении (от x_n до x_0) следующим образом:

- узлу x_n приписывается индекс $\lambda_n = 0$;
- всем узлам (ещё не имеющим индексов), от которых идёт ребро в конечный узел, приписываем индекс $\lambda_i = 1$;
- всем следующим узлам, от которых идут рёбра в только что помеченные узлы, приписываем индекс $(\lambda_i + 1)$.

Этот процесс продолжается концентрическими кругами, расширяясь от x_n , до тех пор, пока не будет помечен начальный узел x_0 графа G_1 . Значение индекса λ_0 у начального узла x_0 и будет равно длине кратчайшего пути.

Пример 2. Пусть дан граф G_1 , представленный на рис. 3.3. Сеть состоит из десяти узлов. Необходимо найти кратчайший путь от x_0 до x_n . На рис. 3.3 показаны значения индексов λ_i для каждого i -го узла. Начинается расчёт узла x_n (для которого $\lambda_n = 0$) и завершается установкой индексов у всех остальных узлов. Индекс при x_0 равен $\lambda_0 = 3$. Поэтому длина кратчайшего пути равна $L_{\text{кр}} = 3$. Само направление кратчайшего пути находим путём составления последовательности узлов по направлению убывания у них индексов от $\lambda_0 = 3$ до $\lambda_n = 0$. Для графа G_1 кратчайший путь составляют узлы x_0, x_1, x_7, x_n . На рис. 3.3 кратчайший путь выделен жирными стрелками.

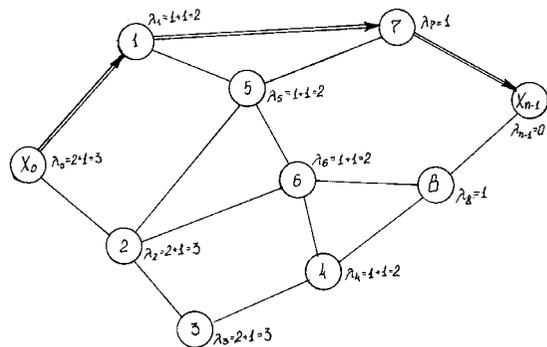


Рис. 3.3. Транспортная сеть равных расстояний

Рассмотрим случай 2, когда длины l_{ij} рёбер (x_i, x_j) транспортной сети различны. Задача усложняется, поскольку часто путь, проходящий через минимальное число вершин, может иметь большую длину (стоимость или время продвижения), чем обходные пути. Алгоритм расчёта индексов похож на предыдущий. Однако есть различия.

Первоначально конечному узлу приписываем индекс $\lambda_n = 0$, а для остальных узлов устанавливаем индекс, равный бесконечно большому числу ($\lambda_i = \infty$).

Ищем такую дугу (x_i, x_j) , для которой справедливо неравенство

$$\lambda_j - \lambda_i > \min_j \{l(x_i, x_j)\} \quad (3.3)$$

и заменяем индекс у x_j узла:

$$\lambda_j = \lambda_i + \min_j \{l(x_i, x_j)\}. \quad (3.4)$$

Продолжаем этот процесс замены индексов аналогично предыдущему случаю до тех пор, пока остаётся хотя бы одна дуга, позволяющая уменьшить индекс λ_j .

Правило нахождения кратчайшего пути при вычисленных индексах λ_i формулируется следующим образом: при обратном проходе по графу G от x_n выбираем тот из узлов, для которого λ_i минимально.

Пример 3. Пусть дан граф G_2 , представленный на рис. 3.4. Сеть состоит из шестнадцати узлов.

Расстояния между узлами различны, и каждое из рёбер графа G_2 помечено величиной расстояния между узлами, соединёнными данным ребром. На рис. 3.4 также указаны значения индексов λ_i для каждого i -го узла. Расчёт начинаем с узла x_n , имеющего индекс $\lambda_n = 0$, и завершаем установкой индексов у всех остальных узлов. Индекс при x_0 равен $\lambda_0 = 26$, что означает $L_{\text{кр}} = 26$. Само направление кратчайшего пути находим аналогично первому случаю путём составления последовательности узлов по направлению убывания у них индексов от $\lambda_0 = 26$ до $\lambda_n = 0$. Для графа G_2 кратчайший путь составляют узлы $x_0, x_5, x_8, x_{11}, x_{13}, x_{n-1}$. На рис. 3.4 кратчай-

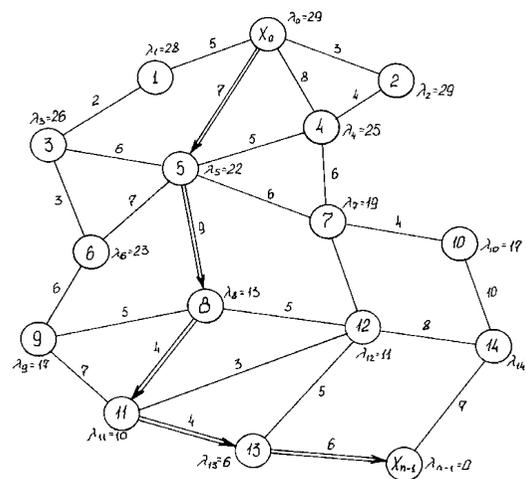


Рис. 3.4. Транспортная сеть с разными расстояниями между узлами

логично первому случаю путём составления последовательности узлов по направлению убывания у них индексов от $\lambda_0 = 26$ до $\lambda_n = 0$. Для графа G_2 кратчайший путь составляют узлы $x_0, x_5, x_8, x_{11}, x_{13}, x_{n-1}$. На рис. 3.4 кратчай-

ший путь также выделен жирными стрелками. Обратим внимание на формирование индексов у x_{12} : $\lambda_{12}=11$ как результат сложения индекса $\lambda_{13}=6$ с длиной $l(x_{13}, x_{12})=5$. Как видим, работает правило минимизации длины возможных путей от x_{12} до x_n .

3.1.3. Построение графа наименьшей длины

Граф наименьшей длины всегда является деревом, поскольку при наличии цикла можно было бы удалить одно из рёбер, и все узлы всё ещё были бы соединены между собой. Используется такой граф обычно при построении сети трубопроводов или линий электропередач. Важен сам факт соединения всех узлов сети, а само расстояние между входом и выходом сети имеет второстепенное значение. С математической точки зрения задача формулируется следующим образом: соединить все $(n+1)$ вершин графа G_3 таким образом, чтобы число рёбер было минимальным и равным n . Идея алгоритма построения заключается в следующем. Пусть заданы координаты узлов будущей сети (x_i) . Просматривая всевозможные расстояния между узлами, выбираем пару вершин x_i и x_j – такую, что длина ребра $l(x_i, x_j)$ у них является минимальной. Затем к этому ребру ищется вершина x_k , у которой длина ребра $l(x_k, x_i)$ или $l(x_k, x_j)$ будет минимальной. Формируют новое ребро будущего графа, подсоединяя узел x_k к тому из узлов, для которого длина подсоединяемого ребра будет минимальной. Необходимо следить, чтобы при этой операции подсоединения узла x_k не образовался цикл. При возможности цикла узел x_k исключается из рассмотрения, и снова ищут новую вершину x_{k+1} . Процесс построения графа G_3 продолжают до тех пор, пока все узлы не будут соединены в сеть. Дерево, построенное таким образом, называется экономическим деревом. Его длина равна сумме соединяющих его рёбер.

Пример 4. Пусть на рис. 3.5 даны координаты (x_1, x_2, \dots, x_9) девяти городов. Необходимо соединить эти города сетью трубопроводов, имеющей минимальную длину. Построим экономическое дерево согласно описанному выше алгоритму. Минимальное расстояние между узлами будет $l(x_5, x_6)$. Теперь ищем узлы, ближе всего отстоящие от вершин x_6 и x_5 . В нашем примере – это узел x_8 . Поэто-

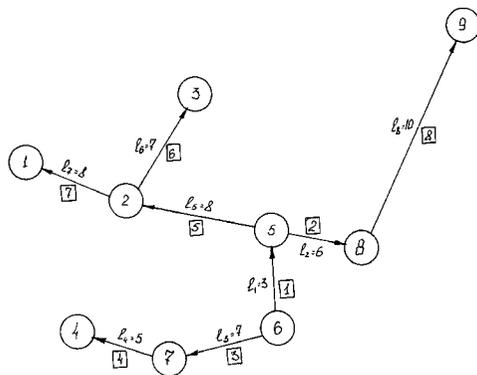


Рис. 3.5. Пример построения экономического дерева

му на втором шаге к имеющемуся ребру (x_5, x_6) добавляем ребро (x_5, x_8) . На третьем шаге к сети, состоящей из двух рёбер, добавится аналогичным образом ребро (x_6, x_7) . На рис. 3.5 каждое ребро сформированной сети помечено номером итерации, на которой оно подсоединено к сети и длиной ребра. Общая длина экономического дерева $L_{\min}=3+6+7+5+8+7+8+10=54$. Как видим, любая другая комбинация соединения девяти узлов при заданном их расположении даст большую длину сети.

3.1.4. Задача об оптимальном назначении

Пусть необходимо выполнить некоторый комплекс независимых друг от друга строительных работ P_1, P_2, \dots, P_m . Имеется ряд организаций-исполнителей J_1, J_2, \dots, J_n , каждая из которых способна выполнить только некоторые из этих работ. Любая из организаций в данный момент времени может выполнять только одну работу. Нужно распределить работу между исполнителями таким образом, чтобы одновременно выполнялось наибольшее количество работ. Эту содержательную постановку задачи будем формализовать, используя графовое представление процесса назначения исполнителей по работам.

Введём матрицу возможной выполнимости работ исполнителями $\|\alpha_{ij}\|$,

$$\text{где } \alpha_{ij} = \begin{cases} 1, & i\text{-я работа выполняется } j\text{-м исполнителем;} \\ 0 & \text{при невозможности выполнения.} \end{cases}$$

Определим булевы переменные ξ_{ij} , которые принимают значения 1 в случае, если i -я работа поручена j -му исполнителю ($i = \overline{1, m}, j = \overline{1, n}$). Переменные ξ_{ij} должны удовлетворять следующим $(n+1)$ ограничениям:

$$\sum_{i=1}^m \xi_{ij} \leq 1, \quad j = \overline{1, n}; \quad \xi_{ij} \leq \alpha_{ij}, \quad (3.5)$$

означающим, что работу P_i можно поручить исполнителю J_j ;

$$\sum_{j=1}^n \xi_{ij} \leq 1, \quad i = \overline{1, m}, \quad (3.6)$$

означающим, что работа поручается только одному исполнителю либо никому не поручается.

Нужно максимизировать следующую целевую функцию:

$$\max \sum_{i=1}^m \sum_{j=1}^n \xi_{ij}. \quad (3.7)$$

В результате мы получили задачу об оптимальном назначении, которую

можно решать как задачу целочисленного ЛП. Мы можем построить сеть, содержащую в себе $(m+n+2)$ вершины. В этой сети m вершин соответствуют работам (P_i) , n вершин – исполнителям (J_j) , и две особые вершины (S' – исток, S'' – сток). Рёбра в этой сети проведём следующим образом: соединим исток S' с P_i и сток S'' с J_j . Поток от P_i к J_j обозначим как ξ_{ij} . Аналогично обозначим $\xi_{S'i}$ и $\xi_{jS''}$. Каждому распределению работ P_i можно поставить в соответствие в только что построенной сети распределение исполнителей. Любому потоку с целочисленными компонентами соответствует некоторое распределение работ. И задача сводится к отысканию максимального потока в транспортной сети, представленной на рис. 3.6.

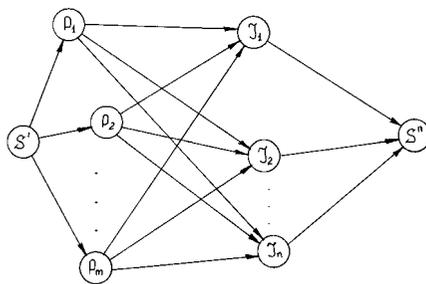


Рис. 3.6. Сеть оптимальных назначений исполнителей.

3.2. Модели сетевого планирования

3.2.1. Технология построения сетевых графиков

В задачах рационального планирования комплекса взаимосвязанных работ используются модели сетевого планирования. Обычно комплекс состоит из ряда элементарных работ, которые взаимосвязаны таким образом, что выполнение некоторых из них не может быть начато раньше, чем закончатся другие работы или не будут выполнены определённые условия. Как правило, количество таких звеньев работ большое, они сложным образом взаимосвязаны, требуют большого числа разнообразных ресурсов и различного времени их исполнения. Всё это требует использования специального математического аппарата, который получил название «методы сетевого планирования для решения прямых и обратных задач ИСО». Прямые задачи ИСО отвечают на вопрос: что будет, если мы примем данную организацию комплекса работ. Обратные задачи ИСО позволяют определить, каким образом нужно организовать все работы, чтобы завершить комплекс работ с максимальной эффективностью. Сетевая модель изображается в виде сетевого графика (СГР), который составляется из стрелок, обозначающих процессы выполнения работ, и кружков, характеризующих свершение событий (результаты выполнения группы взаимосвязанных работ). При этом оперируют тремя понятиями: работа, событие, путь (на сетевом графике).

Работы делятся на три функциональных вида: действительная работа, ожидание, фиктивная работа. Действительная работа представляет собой

процесс, требующий затрат труда, ресурсов и времени. Ожидание требует только затрат времени и может быть обусловлено технологией выполнения работ друг за другом. Например, затвердевание бетона, сушка древесины, обжиг деталей и т.д. Фиктивная работа представляет собой только логическую зависимость между функциональными операциями, представленными действительными работами или ожиданиями. Она не требует ни времени, ни ресурсов, но показывает, что начало одной работы обусловлено либо окончанием другой работы, либо внешними обстоятельствами. Поэтому фиктивную работу вводят для отражения правильной взаимосвязи между работами при построении сетевого графика и изображают пунктирной стрелкой.

Событие – это результат выполнения одной или нескольких работ. Оно не имеет продолжительности. Свершение события означает, что можно приступить к последующим работам. Поэтому факт свершения события является обязательным условием для начала выполнения выходящих из него работ. На сетевом графике выделяют два особых события: исходное и завершающее. Исходному событию не предшествуют работы, а завершающее событие не имеет последующих работ. Каждая работа СГР объединяет два события (начальное и конечное), поэтому обозначается двумя индексами этих событий. Всем событиям на СГР присваиваются порядковые номера i .

Путь в СГР представляет собой любую последовательность работ, в которой конечное событие каждой работы совпадает с начальным событием последующей работы. Путь, начало которого совпадает с исходным событием, а конец которого совпадает с завершающим событием, называется полным. Продолжительность пути L равна сумме продолжительностей составляющих работ. Полный путь, имеющий максимальную длину L , называют критическим, а работы, составляющие этот путь, называются критическими работами. Задержка выполнения любой из критических работ ведёт к более позднему сроку наступления завершающего события. У некритических путей имеются резервы времени, в пределах которых продолжительность отдельных работ может быть увеличена. За счёт этого возможно, например, перераспределение ресурсов от этих работ на критические работы. При построении сетевого графика необходимо соблюдать следующие правила:

- в СГР не должно быть работ, имеющих одинаковые номера;
- все события, кроме завершающего, должны иметь последующую работу. «Тупики» на СГР свидетельствуют либо об ошибке, либо указывают на то, что эта работа является лишней;
- не должно быть событий, в которые не входит ни одна из работ, поскольку условия их свершения не будут обеспечены;
- в СГР не должно быть замкнутых контуров. Поэтому СГР представляет собой специальный вид графа;
- если некоторые работы могут быть начаты после частичного выполне-

ния работ, то предшествующую работу надо разбить на части и ввести дополнительные события;

– если для свершения события кроме работ, указанных на СГР, нужны результаты от посторонних условий, то это изображается с помощью дополнительных надписей.

Процесс разработки СГР разбивается на две стадии: подготовка исходных данных, построение СГР. На первой стадии необходимо выявить все взаимозависимости и установить продолжительности работ. Работы ij и jk называются взаимозависимыми, если одна из них может быть начата только после завершения другой. В противном случае они могут выполняться параллельно. Взаимосвязь между работами обычно представляют в виде матрицы связей (матрицы следования), обозначаемой $\|\alpha_{ij}\|$. В этой матрице, состоящей из нулей или единиц, на пересечении строки j и столбца i ставится $\alpha_{ij}=1$, если работа j может начаться только после завершения работы i . В противном случае эти работы независимы, и $\alpha_{ij}=0$. Как видим, эта матрица $\|\alpha_{ij}\|$ является треугольной снизу.

Вторым шагом первой стадии построения СГР является оценка продолжительности работ. Существуют два подхода к нахождению продолжительности ij -й работы (t_{ij}): детерминистический и вероятностный. При первом подходе t_{ij} представляют собой некоторую нормативную (постоянную) величину. При втором подходе t_{ij} есть случайная величина, но при этом известен закон и параметры её распределения $F(t_{ij})$.

После нахождения всех времён t_{ij} и составления матрицы следования $\|\alpha_{ij}\|$ переходят ко второй стадии: построению СГР. Сам процесс построения может быть либо машинным, либо с помощью графика. Остановимся на графическом представлении СГР. Используются две формы СГР: полигональная и ортогональная. В первом случае работы ij изображаются в виде безмасштабных стрелок, длины которых не зависят от значения t_{ij} , и эти стрелки могут представляться под любым углом к оси абсцисс. Во втором случае работы изображаются в виде горизонтальных стрелок, длина которых пропорциональна времени t_{ij} . В нижней части графика вводится шкала времени t_0 . Для связи параллельных веток СГР между собой используют фиктивные работы, изображаемые пунктирными линиями.

Полигональная форма обычно используется при поиске оптимального варианта СГР, нахождении узких мест и резервов для их перераспределения. Ортогональная форма более удобна в качестве средства контроля за реализацией её выполнения. Например, на строительстве или предприятии все работы всех служб контролируются по СГР, когда в нижней части СГР указаны календарные даты. Невыполнение работ к указанным срокам является сигналом для дополнительных организационно-технических мероприятий

по обеспечению выполнимости работ согласно СГР.

Третья стадия календарного планирования включает в себя расчёт и анализ параметров СГР. В ходе этого анализа вычисляется критический путь и резерв времени выполнения работ, расположенных на других путях. При этом выполняются операции над событиями и работами.

Каждое событие представляется в виде кружка, разбитого на четыре сектора, в которые заносятся характеристики события $s_i = (i, t_i^{\pi}, t_i^p, R_i)$, где i – номер события на СГР, t_i^p – наиболее ранний срок свершения i -го события, t_i^{π} – наиболее поздний срок свершения i -го события, R_i – резерв времени свершения i -го события.

Для каждой работы ij вычисляется четвёрка характеристик: $\rho_{ij} = (t_{ij}^{pn}, t_{ij}^{nn}, t_{ij}^{po}, t_{ij}^{no})$, где t_{ij}^{pn} – раннее начало свершения работы ij , t_{ij}^{nn} – позднее начало выполнения работы ij , t_{ij}^{po} – раннее окончание работы ij , t_{ij}^{no} – позднее окончание выполнения работы ij .

3.2.2. Расчёт параметров сетевого графика при фиксированных временах выполнения работ

Вычисленные значения вектора характеристик свершения событий S_i заносят на СГР, и туда же возле каждой работы заносят вектор характеристик свершения работ p_{ij} и переходят затем собственно к анализу СГР.

На первом этапе рассчитывают наиболее ранние сроки свершения событий t_i^p . Расчёт производится в направлении от исходного к завершающему событию. Для любого события i ранний срок свершения равен длине максимального из путей к данному событию от исходного. Любое i -е событие считается свершившимся лишь в том случае, когда выполнены все работы, входящие в него. Если к событию j сети подходят несколько работ с известными t_i^p , то

$$t_j^p = \max \{t_i^p + t_{ij}\} . \quad (3.8)$$

Практически при расчёте свершения событий последовательно переходят от исходного события к событиям, всё более от него удалённым. Ранний срок исходного события равен 0. Вычисленные значения t_i^p записываются в левый сектор кружка, обозначающего i -е событие на СГР.

На втором этапе рассчитываются поздние сроки свершения событий t_i^n . Расчёт также ведётся последовательно, но уже в обратном порядке, переходя

от завершающего события к исходному. Поздний срок завершающего события равен его раннему сроку. Времена свершения любых других событий определяются из соотношения

$$t_i^n = \min \{t_j^n - t_{ij}\} . \quad (3.9)$$

Это означает, что от позднего срока всех j -х событий, следующих за i -м событием, вычитаются продолжительности работ, связывающих i -е событие с j -м, и выбирается минимальная величина. Расчёт завершается вычислением t_i^n для исходного события.

На третьем этапе рассчитываются резервы времени выполнения событий по формуле

$$R_i = t_i^n - t_i^p . \quad (3.10)$$

Резервы R_i показывают, на какой предельно допустимый период времени можно задержать свершение события i , не изменяя времени свершения завершающего события.

На четвёртом этапе находят значения компонентов векторов p_{ij} по формулам:

$$t_{ij}^{pn} = t_i^p; \quad t_{ij}^{nn} = t_j^n - t_{ij}; \quad t_{ij}^{po} = t_i^p + t_{ij}; \quad t_{ij}^{no} = t_j^n . \quad (3.11)$$

Когда все характеристики оценок свершения событий и оценок времён выполнения работ вычислены и нанесены на СГР, приступают собственно к анализу параметров СГР. Анализ этот также выполняется поэтапно.

Вначале находят критический путь. Для этого выбирают все события i , для которых нет резервов ($R_i=0$). Если через событие с $R_i=0$ проходит несколько путей, то критическим окажется наибольший по продолжительности.

Определяют полный и частный резервы СГР. Полным резервом R_{ij} называют время, в течение которого можно увеличить продолжительность работы t_{ij} , не изменяя при этом критический путь. Он вычисляется из выражения

$$R_{ij} = t_j^n - t_i^p - t_{ij} . \quad (3.12)$$

Если полный резерв времени расходуется на какой-либо работе пути, то все последующие работы уже будут иметь нулевой резерв времени и становятся критическими. Частный (или независимый) резерв времени r_{ij} определяется временем, в течение которого можно продолжить длительность работы t_{ij} , не меняя при этом срока раннего начала последующих работ:

$$r_{ij} = t_j^p - t_i^n - t_{ij} . \quad (3.13)$$

Частный резерв образуется только у работ, непосредственно предшествующих событию j , в котором пересекаются пути разной длины. Он показы-

вает, какая часть полного резерва работы ij может быть использована на продолжительности этой и предшествующих ей работ. При этом изъятие этого резерва не вызывает сокращения полного резерва ни у одной из работ, следующих за событием i . Это означает, что даже не изменяется резерв R_{j+1} ни у одной из работ, следующих за событием j .

Для характеристики напряжённости сроков выполнения работ на СГР используются следующие коэффициенты:

– коэффициент напряжённости работы

$$K_{ij} = \frac{T(L_{\max}) - T'_{кр}(L_{\max})}{T'_{кр} - T'_{кр}(L_{\max})} , \quad (3.14)$$

где $T(L_{\max})$ – продолжительность полного пути максимальной длины, проходящего через работу ij ; $T'_{кр}$ – продолжительность критического пути. На практике $K_{ij} \leq 1$, и его значение показывает насколько свободно можно располагать резервом работы. Он характеризует срочность выполнения ij работы, если отсутствуют дополнительные технологические ограничения. Для всех критических работ $K_{ij} = 1$;

– коэффициент полного резерва

$$K_{ij}^p = \frac{R_{ij}}{T_{кр} - T'_{кр}(L_{\max})} . \quad (3.15)$$

Характеризует продолжительность максимального пути, проходящего через работу ij и не совпадающего с критическим путём.

Пример 5. Определить параметры сетевого графика, представленного на рис. 3.7, при указанных (фиксированных) значениях времени выполнения работ t_{ij} .

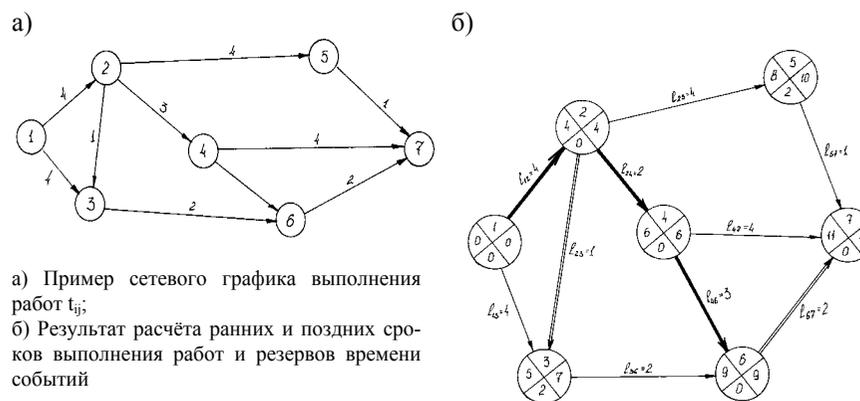


Рис. 3.7. Пример расчёта параметров сетевого графика

При заданных значениях работ рассчитаны ранние сроки свершения событий (в левых секторах кружков, обозначающих события), поздние сроки свершения событий (в правых секторах кружков-событий) и резервы свершения событий (в нижних секторах кружков-событий). Критический путь составляет такая последовательность работ: 1–2; 2–4; 4–6; 6–7.

Длина критического пути $l_{кр} = 4 + 2 + 3 + 2 = 11$. Общее время выполнения комплекса взаимосвязанных работ равно 11. Резервы времени свершения имеют события 3 и 5 (соответственно 2 и 2 единицы времени).

3.2.3. Расчёт параметров сетевого графика при случайных значениях времён свершения работ

В ряде случаев нельзя пренебрегать случайным характером времени выполнения работ t_{ij} , однако исследователю известна функция плотности его распределения $f(t_{ij})$. Если можно предположить независимость длительностей работ на s -м полном пути, то можно найти функцию плотностей $f_s(t)$ как композицию законов распределения $f(t_{ij})$. Затем можно вычислить вероятность выполнения работ на каждом пути за заданное время t_3 по формуле

$$P_s(t_3) = \int_0^{t_3} f_s(t) dt.$$

Критическому пути соответствует минимальное значение этой вероятности. Анализ этого пути, а также путей, имеющих вероятности, близкие к вероятностям выполнения работ критического пути за t_3 , лежит в основе улучшения сетевого графика.

Обычно используют упрощённый метод, основанный на определении средних значений t_{ij} и дисперсий σ_{ij} времён выполнения работ. Перед началом исследований определяются:

- средние значения и дисперсии длительностей работ $(\bar{t}_{ij}, \sigma_{ij}^2)$;
- средние значения и дисперсии наиболее раннего и наиболее позднего сроков свершения каждого события, рассчитанные исходя из средних значений времени выполнения работ $(\bar{t}_i^p, \bar{t}_i^n, \sigma_{t_i^p}^2, \sigma_{t_i^n}^2)$;
- средние значения и дисперсии резервов времени свершения событий $(\bar{R}_i^p, \sigma_{R_i}^2)$;
- вероятность наступления завершающего события на s -м критическом пути к моменту $t_3(P_s(t_3))$.

Отметим, что при определении вероятности свершения завершающего события за t_3 обычно исходят из того, что времена распределены по нормальному закону как сумма большого числа независимых составляющих на

критическом пути:

$$T_{кр} = \sum \bar{t}_{ij}; \quad \sigma^2 = \sum \sigma_{t_{ij}}^2. \quad (3.16)$$

Поэтому вероятность наступления завершающего события определится из выражения

$$P_s(t_3) = \Phi\left(\frac{t_3 - T_{кр}}{\sigma}\right). \quad (3.17)$$

$T_{кр}$ и σ вычисляются по формуле (3.16).

3.2.4. Оптимизация плана комплекса работ

При сетевом планировании исследователи могут преследовать одну из следующих целей.

Во-первых, общее время выполнения комплекса работ T_k руководство предприятия не устраивает. Поэтому ставится задача: каким образом нужно форсировать работы и какие из них, чтобы $T_{кр} = T_3$, где T_3 – некоторое заданное время, удовлетворяющее руководство предприятия. Очевидно, имеет смысл форсировать работы, расположенные на критическом пути СГР. Но критический путь может измениться при перераспределении ресурсов. Тогда критическими окажутся уже другие работы. Любое форсирование работ потребует дополнительных средств. Поэтому возникает задача ИСО: какие необходимы дополнительные средства и в какие работы их вложить, чтобы общий срок выполнения комплекса работ был не больше заданной величины T_3 , а расход этих средств был при этом минимальным.

Во-вторых, может преследоваться цель перераспределения имеющихся резервов ресурсов между работами. Для этого необходимо перебросить силы с некритических участков на критические и за счёт этого добиться уменьшения $T_{кр}$.

В-третьих, анализ СГР показал, что $T_{кр} < T_3$. Это означает, что у руководства предприятия есть резерв времени выполнения некритических работ, их можно растянуть и за счёт этого сэкономить средства на выполнение комплекса работ.

Все три случая являются типичными задачами ИСО. Так, при решении задачи 1 поступают следующим образом. Пусть комплекс работ состоит из a_1, \dots, a_n работ с соответственно временами их выполнения t_1, \dots, t_n . Критический путь известен:

$$T_{кр} = \sum t_i > T_3.$$

Известно, что вложение определённой суммы x дополнительных средств в работу a_i сокращает время с t_i до $t_i = f_i(x)$. Ставится задача: какие дополнительные части средств x_1, x_2, \dots, x_n следует вложить в каждую из работ, чтобы:

- срок выполнения комплекса T удовлетворял бы требованиям руково-

дства:

$$T' = \sum f_i(x_i) \leq T_3;$$

– сумма вложенных средств достигла минимума:

$$X = \sum_{i=1}^n x_i \rightarrow \min .$$

В общем случае поставленная задача относится к классу задач нелинейного программирования. Если же $f_i(x_i)$ – линейная функция, или изменяемые значения небольшие – такие, что их можно считать линейными, то можно прийти к задаче ЛП.

При решении задачи 2 предполагают, что для совокупности работ a_1, \dots, a_n с временами выполнения t_1, \dots, t_n критический путь опять же определяется их суммированием $T_{кр} = \sum t_i$, но на некритических работах есть резервы времени. На предприятии обычно имеется некоторый запас подвижных средств, которые распределяются между работами соответственно b_1, \dots, b_n ; $\sum_{i=1}^n b_i = B$. Известно также, что количество средств $x_i > 0$, снятие с

работы a_i приводит к увеличению времени выполнения работы i с t_i до $t'_i = f_2 |x_i| < t_i$. Количество средств x_i , вложенное в работу a_i , уменьшает время её выполнения до $t''_j = \varphi_j(x_j) < t_j$. При этом величины x_i должны удовлетворять ограничениям $x_i > b_i$. Причём для случая переброски ресурсов с одной работы на другую должно соблюдаться условие $x_1 + x_2 + \dots + x_n = 0$. Для тех работ, на которые перебрасываются ресурсы, $t''_i = \varphi_i(x_i)$, а если из работы снимают ресурсы, то $t'_i = f_i(x_i)$.

В результате таких перебросок общий срок выполнения работ

$$T' = \sum_{кр} \varphi_j(x_j) + \sum_{нкp} f_i(x_i).$$

Снова имеем задачу ИСО: найти такие значения переменных x_i , чтобы выполнить ограничение $x_i \geq b_i$. Это достаточно сложная задача. Даже если $f_i(x_i)$ и $\varphi_j(x_j)$ – линейные функции, возникает задача нелинейного программирования. Существенно нелинейным является то, что значения номеров работ ij , на которые распределяются ресурсы, сами зависят от величины x_i .

Третья задача предполагает поиск варианта перераспределения ресурсов у работ a_1, \dots, a_n с временами t_1, \dots, t_n и за счёт этого увеличение длительности выполнения некоторых из них, поскольку есть резерв времени их выполнения ($T_{кр} < T_3$). За счёт этой операции руководство хочет получить экономию средств. Известно при этом, что доведение t_i до $t_i + \tau_i$ освобождает средства, которые можно вычислить по формуле $x_i = f(\tau_i)$. Нужно найти ве-

личину x_i , на которую можно задержать i -ю работу, чтобы $T_3^- - T_{кр} = 0$ и при этом экономия опять же была минимальной. Мы приходим снова к формулировке задачи нелинейного программирования:

$$\sum_{кр} (t_i + \tau_i) \leq T_3 \text{ при } \sum f_i(\tau) \Rightarrow \max .$$

Успех решения подобных задач зависит от вида и параметров функции плотности распределения $f_i(\tau_i)$.

Методики решения подобных задач рассматриваются во второй части данного пособия.

3.3. Примеры применения графовых моделей и моделей сетевого планирования

3.3.1. Задача оптимизации документооборота

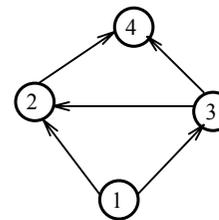


Рис. 3.8. Информационный граф документооборота

Пусть модель документооборота предприятия задана информационным графом (рис 3.8).

Необходимо оптимизировать документооборот.

Решение. Построим матрицу смежности $A = |a_{ij}|$:

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

где строки i и столбцы j соответствуют вершинам графа, 0 – нет прямой связи с другими вершинами, 1 – есть связь.

Алгоритм.

Шаг 1. Расчёт последовательности матриц

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}; A^2 = \begin{bmatrix} 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}; A^3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

$$A^4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 2 & 1 & 3 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Элемент a_{ij}^λ указывает количество различных путей длины λ из x_i в x_j .

Матрица B называется матрицей достижимости, элемент b_{ij} указывает общее число путей различной длины из x_i в x_j . Если в какой-то A_i на диагонали появился не нуль, то это указывает на наличие контура в графе (т.е. документ необходим для формирования другого документа и в то же время не может быть сформирован, пока не сформируется последний), и, следовательно, существует ошибка обследования потока документов в системе.

Шаг 2. По матрице смежности определяем $p_j = \sum_i a_{ij}$ и $q_i = \sum_j a_{ij}$:

$$\bar{p} = (0, 2, 1, 2); \bar{q} = (2, 1, 2, 0).$$

Если $p_j = 0$, то данный документ является начальным, если $q_i = 0$, то это конечный документ.

$p_j > 0$ – количество документов, используемых при составлении x_j ;

$q_i > 0$ – количество документов, при составлении которых необходима непосредственная информация из x_i .

Шаг 3. Определение первичных и производных документов. Если $b_{ij} \neq 0$, то документ x_i используется при построении x_j (необязательно непосредственно). Таким образом, ненулевые элементы j -го столбца матрицы B определяют все первичные для x_j документы, а ненулевые элементы i -й строки – все производные от x_i документы.

Шаг 4. Определяется порядок элементов $x_j - \pi_j$ путём расчёта величин $\sigma_j = \sum_i a_{ij}^{(n)}$:

$$\bar{\sigma}(1) = (0, 2, 1, 2);$$

$$\bar{\sigma}(2) = (0, 1, 0, 3);$$

$$\bar{\sigma}(3) = (0, 0, 0, 1);$$

$$\bar{\sigma}(4) = (0, 0, 0, 0).$$

За π_j принимается λ^* , при котором

$$\begin{cases} \bar{\sigma}(\lambda^*) > 0; \\ \bar{\sigma}(\lambda^* + 1) = 0. \end{cases}$$

Имеем $\bar{\pi} = (0, 2, 1, 3)$.

Величина π_j определяет номер такта подготовки документов, к которому готовы все исходные документы для x_j .

Величины $k = \max_j \pi_j$ называются порядком информационного графа ($k=3$). Такты нумеруются с нуля.

Шаг 5. Номер такта, после которого документ x_i может быть сдан в архив

или уничтожен, определяется следующим образом:

$$\tau_i = \max_j \{ \pi_j a_{ij} \}.$$

Если документ конечный, то $\tau_i = \pi_i$. Имеем: $\bar{\tau} = (2, 3, 3, 3)$.

Шаг 6. Минимальное число тактов, в течение которых документ необходимо хранить в подразделениях предприятия, $\theta_i = \tau_i - \pi_i$, $\bar{\theta} = (2, 1, 2, 0)$.

Шаг 7. Наличие в матрице B элементов, отличных от 0 и 1, указывает, что информация из одного документа поступает в другой несколькими путями. Это свидетельствует о формальной избыточности информационного графа и указывает на необходимость содержательного анализа потоков информации.



Рис. 3.9. Вариант исключения лишних связей и документов из информационного графа

В результате этого анализа могут быть исключены лишние связи и промежуточные документы (рис 3.9).

3.3.2. Моделирование стационарной технологии с помощью моделей сетевого планирования

Существенный элемент технологического процесса работы сортировочной станции – технологические графики обработки поездов и вагонов в её цехах: парках прибытия и отправления, на горке и т. д. Каждый график точно определяет порядок и последовательность всех операций в рамках определённой части технологии. Однако представить по ним технологическую линию затруднительно. Сетевой график более нагляден. Сетевой график подвергается анализу с целью разработки мероприятий для совершенствования технологии. Далее строят варианты сетевого графика, из которых выбирают наилучший. Этот процесс называется **оптимизацией сетевого графика**.

Постановка задачи. Рассмотрим технологическую линию переработки транзитных вагонов на сортировочной станции. Построить укрупнённый сетевой график, отражающий технологию обработки вагонов замыкающей группы, прибывшей с поездом, на который телеграмма – натурный лист не получена. Документы на станции пересылают с рассыльным. На станции имеется единая техническая контора (ТК) и механизированная горка. Исходное событие графика – момент прибытия в расформирование поезда с замыкающей группой (событие 1), завершающее – готовность сформированного поезда к отправлению (событие 14). Перечень работ дан в табл. 3.1.

Для заданного сетевого графика комплекса работ по переработке транзитных вагонов на сортировочной станции определить:

- 1) плановое время начала каждой работы;
- 2) критический путь и его длину;

3) полные, свободные и независимые резервы времени выполнения работ.

Таблица 3.1. Перечень работ сетевого графика.

| <i>i</i> | <i>j</i> | <i>t_{ij}</i> |
|----------|----------|-----------------------|----------|----------|-----------------------|----------|----------|-----------------------|----------|----------|-----------------------|
| 1 | 2 | 15 | 4 | 5 | 5 | 7 | 8 | 15 | 10 | 13 | 23 |
| 1 | 3 | 15 | 4 | 9 | 10 | 8 | 9 | 15 | 11 | 12 | 5 |
| 2 | 4 | 10 | 5 | 6 | 10 | 8 | 10 | 5 | 12 | 13 | 15 |
| 3 | 5 | 5 | 6 | 7 | 10 | 9 | 11 | 8 | 13 | 14 | 2 |

Построим сетевой график и проложим критический путь (рис. 3.10).

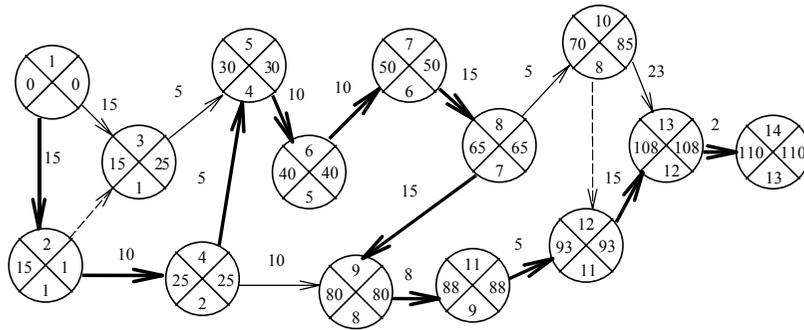


Рис. 3.10. Сетевой график комплекса работ

Критический путь выражает минимально возможный простой перерабатываемого на станции вагона без времени накопления, т. е. простой вагона замыкающей группы.

Таким образом, задача сводится к сокращению критического пути. Этого можно добиться, во-первых, логической перестройкой сети так, чтобы работы, лежащие на нём, выполнялись не последовательно, а параллельно (совершенствование методов работы), во-вторых, сокращением продолжительности работ, лежащих на критическом пути (применение более совершенных технических средств, механизация, автоматизация процессов и т. д.). Первый путь требует организационных мер, второй – капитальных вложений.

В левом секторе исходного события 1 ставим ноль. Переходим ко второму событию. В него входит только работа (1, 2) продолжительностью $t_{12}=15$, поэтому в левом секторе второго события запишем 15 ($0+15=15$), в нижний записываем 1 (номер события, от которого был сделан переход).

Далее левые и нижние секторы событий будут заполнены следующим образом:

$$\begin{aligned}
 T_{E3} &= 0 + 15 = 15; k = 1; & T_{E6} &= \max(30 + 10) = 40; k = 5; \\
 T_{E4} &= 15 + 10 = 25; k = 2; & T_{E7} &= \max(40 + 10) = 50; k = 6; \\
 T_{E5} &= \max(15 + 5; 25 + 5) = 30; k = 4; & T_{E8} &= 50 + 15 = 65; k = 7;
 \end{aligned}$$

$$\begin{aligned}
 T_{E9} &= \max(25 + 10; 65 + 15) = 80; k = 8; & T_{E12} &= 88 + 5 = 93; k = 11; \\
 T_{E10} &= 65 + 5 = 70; k = 8; & T_{E13} &= \max(70 + 23; 93 + 15) = 108; \\
 T_{E11} &= 80 + 8 = 88; k = 9; & & k = 12; \\
 & & T_{E14} &= 108 + 2 = 110; k = 13.
 \end{aligned}$$

Двигаясь по графу в обратном направлении, получим

$$\begin{aligned}
 T_{L14} &= 110; & T_{L7} &= 50; \\
 T_{L13} &= 110 - 2 = 108; & T_{L6} &= 40; \\
 T_{L12} &= 108 - 15 = 93; & T_{L5} &= 30; \\
 T_{L11} &= 88; & T_{L4} &= 25; \\
 T_{L10} &= 85; & T_{L3} &= 15. \\
 T_{L9} &= 80; & T_{L2} &= 15; \\
 T_{L8} &= 65; & T_{L1} &= 0.
 \end{aligned}$$

Определение критического пути ведется от последнего события сети. В качестве события, предшествующего номеру *j* в критическом пути, принимается событие с номером *k*, для которого $T_{Ek} + t_{kj} = \max_i [T_{Ei} + t_{ij}]$.

Критический путь: 1-2-4-5-6-7-8-9-11-12-13-14.

Его длина: $T_{кр.} = T_{E14} = T_{L14} = 110$.

Резерв времени R_i для событий, лежащих на критическом пути, $R_i = 0$, т.к. $T_{Ei} = T_{Li}$.

$R_i = T_{Li} - T_{Ei}$ показывает, на какое время можно задержать событие, не изменяя общего срока выполнения процесса.

Для работ, не лежащих на критическом пути, резервы времени выполнения работ вычислим по формулам:

$$\text{полные: } R_{ij} = T_{Lj} - (T_{Ei} + t_{ij});$$

$$\text{свободные: } R_{ij}^c = T_{Ej} - (T_{Ei} + t_{ij});$$

$$\text{независимые: } R_{ij}^n = T_{Ej} - (T_{Li} + t_{ij}).$$

Резервы времени выполнения работ, не лежащих на критическом пути, занесём в табл. 3.2.

Таблица 3.2. Резервы времени выполнения работ

| <i>i-j</i> | R_{ij} | R_{ij}^c | R_{ij}^n |
|------------|----------|------------|------------|
| 1-3 | 10 | 0 | 0 |
| 3-5 | 10 | 10 | 0 |
| 4-9 | 45 | 45 | 45 |
| 8-10 | 15 | 0 | 0 |
| 10-13 | 15 | 15 | 0 |

4. МОДЕЛИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

4.1. Идея метода динамического программирования

Динамическое программирование представляет собой вычислительный метод для решения задач определённой структуры. Предложен метод Р. Беллманом в 1953 году для задач управления запасами. Сущность метода рассмотрим на следующем примере. Пусть необходимо исследовать производственный процесс (ППР), который распадается на n шагов или этапов. При этом показатель эффективности производства W_n складывается из составляемых W_j , полученных на отдельных этапах: $W_n = \sum_{j=1}^n W_j$. На каждом j -м

этапе W_j (например, доход от ТПР) вычисляется по итеративному алгоритму. Для каждого j -го этапа существует своя система воздействий на ТПР, обеспечивающая оптимум W_j на этапе j , называемая в дальнейшем условным управлением ($УП_j$). Для решения такого рода задач составляются ММ динамического программирования. При этом весь плановый период ППР (T), называемый в дальнейшем интервалом оптимизации (который в ММ ЛП ранее был неизменен, и расчёт вёлся на конец периода T), теперь уже разбивается на n этапов. Поэтому появляется необходимость в динамическом расчёте этого показателя W_j . Отсюда заимствован термин динамического программирования. Это направленный последовательный перебор вариантов, который приводит к поиску глобального экстремума. Рассмотрим несколько примеров задач динамического программирования.

Пример 1. Планируется деятельность группы промышленных предприятий $П_1, П_2, \dots, П_k$ на период в m лет. В начале каждого года суммарные средства заново перераспределяются. Ставится задача: определить количество средств в начале каждого года, которые нужно выделить каждому предприятию, чтобы максимизировать суммарный доход за m лет.

Пример 2. Владелец машины эксплуатирует её в течение m лет. В начале каждого года он может принять решение по годам (продать, отремонтировать, эксплуатировать), для того чтобы сумма расходов (на эксплуатацию, ремонт и приобретение) была минимальной.

Пример 4.3. Предприятие выпускает продукцию, на которую имеется различный спрос по годам (D_t). В начале t -го планового периода ($t=1, 2, \dots, N$) у предприятия образуется различный уровень запасов (i_t). Нужно определить размер выпуска продукции X_t в течение периода t . Главным условием решения задачи является целочисленность объёма выпуска ($X_t=0, 1, 2, \dots$) и

интервалов деления общего планового периода T ($t=1, 2, \dots, N$). На конец планового периода ($t=N$) уровень запасов предприятия должен быть равен нулю ($i_N=0$), что соответствует полной реализации продукции предприятия. Известна также функция затрат на выпуск и хранение продукции, которая является нелинейной и различной для каждого интервала $C_t(X_t, i_t)$. Расчёт запасов ведётся по итеративной формуле

$$i_t = i_{t-1} + X_t - D_t, \quad (4.1)$$

которая означает условие полного и своевременного удовлетворения спроса в пределах t -го временного интервала. Зависимость (4.1) означает равенство уровня запасов к концу t -го периода уровню запасов на начало периода, складываемому с объёмом выпуска и вычитаемому спросу на продукцию.

Общим для перечисленных задач является многошаговый характер оптимизации. Осуществляется замена решения n -шаговой задачи последовательностью задач: одношаговой, двушаговой и т.д. Для применения метода динамического программирования задачи производства должны обладать следующими свойствами:

1. Задача должна допускать интерпретацию как n -шаговый процесс принятия решений.
2. Задача должна быть определена для любого числа шагов и иметь структуру, не зависящую от их числа.
3. При рассмотрении k -шаговой задачи должно быть задано некоторое множество параметров, описывающих состояние системы, от которых зависят оптимальные значения переменных. Причём это множество не должно изменяться при увеличении числа шагов.
4. Выбор решения ($УП_j$) на j -м шаге не должен оказывать влияние на предыдущие решения, кроме необходимого пересчёта переменных.

Пусть вектор Θ – вектор параметров, описывающих состояние ПТР (вектор состояния), тогда $\Lambda_j(\Theta)$ – оптимальное значение целевой функции для j -шагового ТПР будем называть функцией состояния. Обозначим через X_j вектор переменных управления, подлежащих выбору на j -м шаге. Для задач, к которым можно применять метод динамического программирования, должно выполняться следующее основное рекуррентное соотношение:

$$\Lambda_j(\Theta) = \max_{X_j} \{ f(X_j, \Theta) + \Lambda_{j-1}[T(\Theta, X_j)] \}, \quad (4.2)$$

где $T[\Theta, X_j]$ – вектор состояния предыдущего ($j-1$)-го шага при условиях Θ и X_j . Принцип оптимальности Р. Беллмана, обосновывающий рекуррентное соотношение (4.2), формулируется следующим образом. Оптимальная стратегия обладает свойством: каковы бы ни были начальное состояние и начальная стратегия X_1 , последующие стратегии (решения) должны быть оптимальны по отношению к текущему состоянию системы Θ . Таким образом,

принцип оптимальности утверждает, что оптимальное управление системой на каждом шаге не зависит от предыстории процесса, а определяется только самим этим состоянием. Известно, что системы (процессы), для которых справедливо это свойство, называются марковскими.

Процесс динамического программирования разворачивается от конца к началу:

- раньше всего планируется последний $(n-1)$ -й шаг;
- для каждого из предположений (вариантов исследований) находим условное оптимальное управление (U_{n-1}) и условный оптимальный выигрыш на оставшихся двух последних шагах (W_{n-1}) ;
- далее оптимизируется управление на $(n-2)$ -м шаге (U_{n-2}) и условный оптимальный выигрыш на оставшихся трёх шагах (W_{n-2}) ;
- таким образом поступаем до тех пор, пока не дойдём до первого шага.

В этом случае условный оптимальный выигрыш W_1 становится глобальным ($W_n = W_1$). Для нахождения программы достижения W_n осуществляется второй цикл оптимизации (от начала к концу). На каждом шаге, начиная с первого, выбирают из всех возможных вариантов тот, у которого U_j соответствует условному оптимуму. В результате программа оптимизации представляет собой последовательность условных оптимальных управлений (U_1, U_2, \dots, U_n) .

Вернёмся к примеру 4.3. Для вывода итеративной формулы вида (4.2) для расчёта оптимального значения выпуска продукции на каждом периоде (пусть их будет четыре: $t=1, 2, 3, 4$) составим матрицу ограничений (табл. 4.1).

Таблица 4.1. Матрица ограничений

| Периоды | Параметры ТПР | | | | | | | Спрос |
|---------|---------------|-------|-------|-------|-------|-------|-------|-------------|
| | x_1 | i_1 | x_2 | i_2 | x_3 | i_3 | x_4 | |
| 1 | 1 | -1 | | | | | | $D_1 - i_0$ |
| 2 | | 1 | 1 | -1 | | | | D_2 |
| 3 | | | | 1 | 1 | -1 | | D_3 |
| 4 | | | | | | 1 | 1 | D_4 |

Обозначим: $f_n(i)$ – стоимость затрат на n оставшихся интервалах, соответствующая стратегии их минимизации; $x_n(i)$ – планируемый выпуск продукции на i -м интервале, который обеспечивает достижение затрат $f_n(i)$. В этой матрице 1 означает присутствие в итеративной формуле на i -м шаге данного компонента со знаком “плюс”, а -1 предполагает вычитание компоненты в i -й итерации. Итеративная формула может быть получена по индукции:

$$f_n(i) = \min_x [C_n(x_i, i + x_i - dn_i) + f_{n-1}(i + x_i - dn_i)]; \quad i = \overline{0, N}; \quad dn_i < x_i < N \cdot i.$$

4.2. Задача о выборе маршрута движения транспортного средства

Пусть транспортное средство находится на плоскости в точке с координатами (x_0, y_0) . Необходимо выбрать маршрут его движения по пересечённой местности до точки с координатами (x_k, y_k) . При этом делается одно допущение, что транспортное средство может перемещаться либо только по координате x , либо только по координате y . Вся плоскость разбита на ряд элементарных площадок. Известен расход горючего, затрачиваемый транспортным средством на каждой элементарной площадке при движении его вдоль этой площадки по осям координат. Обозначим через C_{ij} расход горючего при передвижении транспортного средства от i -й к j -й точке вдоль площадки ij . Предполагается, что вся область определения задаётся матрицей расхода горючего $\|c_{ij}\|$, $i = \overline{1, n_1}$ – номера интервалов сетки по оси абсцисс,

$j = \overline{1, n_2}$ – номера интервалов сетки по оси ординат. Изменение координат движения транспортного средства может быть различно по осям X и Y . Шаги изменения положения транспортного средства (ТС) по координатам одинаковы: $\Delta x = \frac{x_k - x_0}{n_1}$; $\Delta y = \frac{y_k - y_0}{n_2}$.

Общее число итераций изменения координат при движении ТС от (x_0, y_0) до (x_k, y_k) равно $n = n_1 + n_2$. Расход горючего ТС на l -м шаге обозначим через W_l . При этом на l -м шаге может быть различное число вариантов. Например, в конечной точке (x_k, y_k) $W_0 = 0$ при $l=0$ (нет расхода горючего, поскольку ТС уже находится в конечной точке). При $l=1$ имеем два варианта, у каждого из которых имеется по единственному (оптимальному) решению. При $l=2$ имеем уже 3 варианта, из которых в качестве условного оптимума выбираем тот, который удовлетворяет условию (4.2).

На рис 4.1 приведён пример распределения расхода горючего ТС по заданной координатной сетке изменения движения ТС. В узлах этой координатной сетки изображены кружки, в которые заносятся значения условного оптимума расхода горючего, необходимого при движении ТС из данного узла в конечный узел с координатами (x_k, y_k) . Стрелками будем обозначать условные оптимальные управления, означающие для данной задачи одно из возможных направлений движения ТС: либо по оси x (горизонтальная стрелка \rightarrow), либо по оси y (вертикальная стрелка \uparrow). В результате работы итеративного алгоритма необходимо найти $W_l = W_n$ (минимальный расход горючего для передвижения ТС от узла (x_0, y_0) до узла (x_k, y_k)) и программу реализации этого движения, то есть последовательность горизонтальных и вертикальных стрелок, обеспечивающих расход горючего W_n .

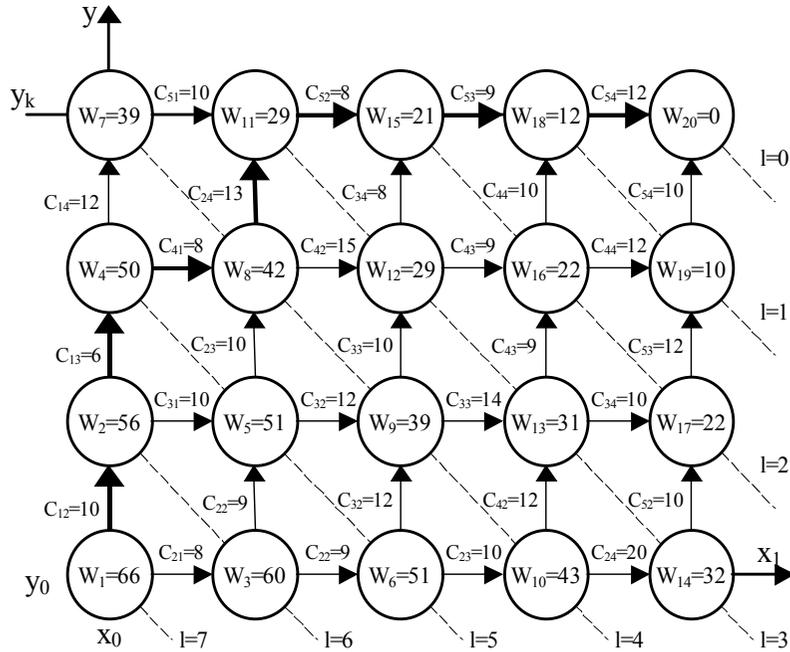


Рис. 4.1. Распределение расхода горючего транспортного средства

На первой итерации ($l=0$) имеем $W_{20}=0, U_{20}=0$.

На второй итерации ($l=1$) $W_{18}=12+0=12, U_{19}=\rightarrow, W_{19}=10+0=10, U_{19}=\uparrow$.

На третьей итерации ($l=2$) $W_{15}=9+12=21, U_{15}=\rightarrow, W_{17}=10+12=22, U_{17}=\uparrow, W_{16}=\min\{10+12; 12+10\}=22, U_{16}=\rightarrow$.

Здесь по обоим направлениям одинаковый расход горючего. Поэтому выбираем любое из направлений. Например, пусть это будет $U_{16}=\rightarrow$.

На четвёртой итерации ($l=3$) поступаем аналогичным образом.

Число вариантов возрастает до четырёх. $W_{11}=8+21=29, U_{11}=\rightarrow, W_{12}=\min\{9+22; 8+21\}=29, U_{12}=\uparrow, W_{13}=\min\{10+22; 9+22\}=31, U_{13}=\uparrow, W_{14}=10+22=32, U_{14}=\uparrow$.

Этот процесс продолжается до шестого шага ($l=6$). На последней итерации $W_n=W_1=\min\{8+60; 10+56\}=66, U_1=\uparrow$. Значение W_n становится общим оптимумом. Первой командой в программе достижения этого оптимума будет $U_1=\uparrow$. Далее процесс поиска программы продолжаем на направлении от начального шага к конечному по направлению, указанному стрелками.

Итак, программа достижения минимума расхода горючего при движении

ТС от узла (x_0, y_0) к узлу (x_k, y_k) имеет вид

$$\uparrow, \uparrow, \rightarrow, \uparrow, \rightarrow, \rightarrow, \rightarrow.$$

Как видим, общее число итераций $n=n_1+n_2=4+3=7$. При выполнении данной последовательности из семи управляющих воздействий (направлений движения ТС по координатной сетке) обеспечивается минимальный расход горючего $W_n=66$ единиц при заданной матрице покоординатного расхода горючего ТС.

4.3. Задача об использовании рабочей силы на предприятии

Существует много задач, где решения должны определяться в строгой временной последовательности. Подобного рода задачи называют задачами последовательного принятия решений. В качестве примера решения подобных задач рассмотрим следующую постановку задачи. Производителю работ нужно определить оптимальное число работников в каждый из n месяцев. Производственные задания для каждого месяца известны. Допустим, что в j -й месяц идеальное число работающих — m_j . Если бы производитель работ мог увольнять и принимать новых рабочих без дополнительных затрат, то он мог бы в j -й месяц принять ровно m_j рабочих ($j=1, 2, \dots, n$).

Предположим, что работа j -го месяца может быть выполнена меньшим числом рабочих при сверхурочной их работе. Необходимо определить фактическое число в j -й месяц x_j . Затраты по изменению численности рабочих при переходе от $(j-1)$ -го месяца к j -му определяются функцией $f_j(x_j-x_{j-1})$. В зависимости от знака величины x_j-x_{j-1} эта функция определяет затраты по найму-увольнению. Очевидно, что $f_j(0)=0$. Отклонение численности рабочих от m_j приводит к расходам $y_j(x_j-m_j), g_j(0)=0, j=1, n$. Считается, что в начальный момент число рабочих составило m_0 . Целевая функция задачи определяется соотношением

$$W = \sum_{j=1}^n [f_j(x_j-x_{j-1}) + g_j(x_j-m_j)], \quad (4.3)$$

где $x_0=m_0$. Имеем задачу с фиксированным началом x_0 . Основное рекуррентное соотношение имеет вид:

$$W = \min_{x_1, \dots, x_n} \{f_1(x_1-m_0) + g_1(x_1-m_1) + \sum_{j=1}^n [f_j(x_j-x_{j-1}) + g_j(x_j-m_j)]\}.$$

Данную задачу удобно решать в обратном направлении. Обозначим

$$\Lambda_n(\Theta) = \min_{x_n} [f_n(x_n-\Theta) + g_n(x_n-m_n)]; \quad \Theta = x_{n-1}.$$

Основное рекуррентное соотношение имеет вид:

$$\Lambda_k(Q) = \min_{x_k} [f_k(x_k - \Theta) + g_k(x_k - m_k) + \Lambda_{k+1}(x_k)], \quad (4.4)$$

где $\Lambda_k(\Theta)$ – минимальные затраты за месяц от k -го до n -го включительно, если количество работников в $(k-1)$ -й месяц равно Q . На последнем шаге определяется оптимальное число рабочих в первый месяц при условии, что на начало месяца их численность составляла m_0 . Определив x_1^* , последовательно находим $x_2^* = x_2(x_1^*)$, $x_3^* = x_3(x_2^*)$. Здесь решение в обратном направлении удобно, так как ничего не известно о числе рабочих на $(n+1)$ -й месяц, тогда как в начале процесса задано m_0 .

Пусть кроме m_0 задано ещё m_{n+1} . Будем искать целые числа, обращающие в минимум выражение

$$W = \sum_{j=1}^n [f_j(x_j - x_{j-1}) + g_j(x_j - m_j) + f_{n+1}(m_{n+1} - x_n)].$$

Перепишем это условие в виде

$$W = \min_{x_n} \{f_{n+1}(m_{n+1} - x_n) + \min_{x_1, \dots, x_{n-1}} \sum_{j=1}^n [f_j(x_j - x_{j-1}) + g_j(m_j - x_j)]\}.$$

Поскольку задано конечное состояние системы m_{n+1} , то будем решать задачу в прямом направлении. Определим последовательность функций состояния:

$$\Lambda_k(\Theta) = \min_{x_1, \dots, x_k} \{f_k(\Theta - x_k) + g_{k+1}(\Theta - m_{k+1}) + \sum_{j=1}^k (f_j(x_j - x_{j-1}) + g_j(m_j - x_j))\}; k = \overline{1, n}.$$

Отсюда

$$\Lambda_1(\Theta) = \min \{f_2(\Theta - x_1) + g_2(\Theta - m_2) + f_1(x_1 - m_0) + g_1(m_1 - x_1)\}.$$

И основное рекуррентное соотношение для данной задачи динамического программирования имеет следующий вид:

$$\Lambda_k(\Theta) = \min_{x_k} \{f_{k+1}(\Theta - x_k) + g_{k+1}(\Theta - m_{k+1}) + \Lambda_k(x_k)\}. \quad (4.5)$$

Функция $\Lambda_k(\Theta)$ представляет собой минимальные затраты в течение первых k месяцев при условии, что численность рабочих в $(k+1)$ -й месяц равна Θ . На последнем шаге при $k=n$, положив $\Theta = m_{k+1}$, приходим к соотношению

$$\Lambda_n(m_{n+1}) = \min_{x_n} \{f_{n+1}(m_{n+1} - x_n) + \Lambda_{n+1}(x_n)\}. \quad (4.6)$$

Определив из (4.6) оптимальное значение x_n^* по таблице предыдущего,

$(n-1)$ -го, шага, найдём искомые значения всех остальных переменных x_{n-1}^* , x_{n-2}^* , ..., x_1^* . Для этого подставим значение $\Theta = x_n^*$.

Итак, если ставится задача последовательного принятия решений на n периодов и задано начальное состояние системы, то задача решается методом динамического программирования в обратном направлении. Если же задано ещё и конечное состояние системы, то решение получают путём динамического программирования в прямом направлении.

Пример 4. Необходимо рассчитать использование рабочей силы на предприятии за четыре периода ($j=1, 2, 3, 4$) при следующем распределении работников: $m_1=2, m_2=5, m_3=3, m_4=1, m_0=2$. Функции $f_j(x_j)$ и $g_j(x_j)$ имеют следующий вид:

$$f_j(x_j) = \begin{cases} 10(x_j - x_{j-1}), & \text{если } x_j > x_{j-1}; \\ 7(x_{j-1} - x_j), & \text{если } x_{j-1} > x_j; \\ f_j(0) = 0; \end{cases} \quad g_j(x_j) = \begin{cases} 8(x_j - m_j), & \text{если } x_j > m_j; \\ 7(m_j - x_j), & \text{если } m_j > x_j; \\ g_j(0) = 0. \end{cases}$$

Поскольку зафиксировано начало ($x_0 = m_0 = 2$), то задачу будем решать в обратном направлении. Основное рекуррентное соотношение

$$\Lambda_k(\Theta) = \min_{x_k} \{f_k(x_k - \Theta) + g_k(x_k - m_k) + \Lambda_{k+1}(x_k)\}, \quad (*)$$

$$\Lambda_n(\Theta) = \min_{x_n} \{f_n(x_n - \Theta) + g_n(x_n - m_n)\}; \quad \Theta = x_{k-1}; \quad n = 1.$$

Первый шаг. $\Lambda_4(\Theta) = \min_{x_4} \{f_4(x_4 - \Theta) + g_4(x_4 - m_4)\} = \min_{x_4} \Pi(x_4, \Theta)$.

Составим таблицу возможных значений $\Lambda_4(\Theta)$ (табл. 4.2).

Таблица 4.2. Возможные значения $\Lambda_4(\Theta)$

| Θ | x_4 | $\Omega(\Theta, x_4)$ | Θ | x_4 | $\Omega(\Theta, x_4)$ |
|----------|-------|-----------------------|----------|-------|-----------------------|
| 0 | 0 | 11 | 3 | 0 | 32 |
| | 1 | 10 | | 1 | 14 |
| | 2 | 28 | | 2 | 15 |
| 1 | 0 | 18 | 4 | 0 | 39 |
| | 1 | 0 | | 1 | 21 |
| | - | - | | - | 22 |
| 2 | 0 | 25 | 5 | 0 | 46 |
| | 1 | 7 | | 1 | 28 |
| | 2 | 8 | | 2 | 29 |

Значения $\min \Omega(x_4, \Theta) = \Lambda_4(\Theta)$ для всех Θ выберем из этой таблицы и сформируем новую таблицу (табл. 4.3).

Таблица 4.3. Минимальные значения $\Lambda_4(\Theta)$

| Θ | Λ_4 | x_4 | Θ | Λ_4 | x_4 |
|----------|-------------|-------|----------|-------------|-------|
| 0 | 10 | 1 | 3 | 14 | 1 |
| 1 | 0 | 1 | 4 | 21 | 1 |
| 2 | 7 | 1 | 5 | 28 | 1 |

Поскольку все функции $f_j(x_j - x_{j-1})$ и $g_j(x_j - x_{j-1})$ выпуклы по переменным x_j , то и функция $\Omega(\xi, x_4)$ также выпукла по x_4 для всех Θ .

Поэтому для нахождения $\Lambda_4(\Theta)$ достаточно определить первый относительный минимум функции $\Omega(x_4, \Theta)$ по x_4 , который и будет глобальным. В соответствии с этим значения функции $\Omega(x_4, \Theta)$ вычисляем лишь для области значений x_4 , содержащей этот минимум, следующим образом:

1. $\Theta = 0; x_4 = 2; \Omega_4 = 0 + 11(1 - 0) = 11; \Theta = 0; x_4 = 1; \Omega_4 = 10(1 - 0) + 0 = 10; \Theta = 0; x_4 = 0; \Omega_4 = 10(2 - 0) + 8(2 - 1) = 28.$

2. $\Theta = 1; x_4 = 0; \Omega_4 = 7(1 - 0) + 11(11 - 0) = 7 + 11 = 18; \Theta = 1; x_4 = 1; \Omega_4 = 0 + 0 = 0.$

3. $\Theta = 2; x_4 = 0; \Omega_4 = 7(2 - 1) + 11(1 - 0) = 25; \Theta = 2; x_4 = 1; \Omega_4 = 7(2 - 1) + 0 = 7; \Theta = 2; x_4 = 2; \Omega_4 = 7 \cdot 0 + 8(2 - 1) = 8.$

4. $\Theta = 3; x_4 = 0; \Omega_4 = 7(3 - 0) + 11(1 - 0) = 32; \Theta = 3; x_4 = 1; \Omega_4 = 7(3 - 1) + 0 = 14; \Theta = 3; x_4 = 2; \Omega_4 = 7(3 - 2) + 8(2 - 1) = 15.$

5. $\Theta = 4; x_4 = 0; \Omega_4 = 7(4 - 0) + 11(1 - 0) = 39; \Theta = 4; x_4 = 1; \Omega_4 = 7(4 - 1) + 0 = 21; \Theta = 4; x_4 = 2; \Omega_4 = 7(4 - 1) + 8(2 - 1) = 22.$

6. $\Theta = 5; x_4 = 0; \Omega_4 = 7(5 - 0) + 11(1 - 0) = 46; \Theta = 5; x_4 = 1; \Omega_4 = 7(5 - 1) + 0 = 28; \Theta = 5; x_4 = 2; \Omega_4 = 7(5 - 2) + 8 = 29.$

Все вычисленные значения Ω_4 занесены в табл. 4.4.

Второй шаг. Используем основное рекуррентное соотношение

$$\Lambda_3(\Theta) = \min_{x_3} \{f_3(x_3 - \Theta) + g_3(x_3 - m_3) + \Lambda_4(x_3)\}$$

и проведём вычисления аналогично первому шагу. Результаты вычислений представлены в табл. 4.4.

Значения $\min \Omega_3 = \Lambda_3(\Theta)$ для всех Θ выберем из табл. 4.4 и сформируем новую (табл. 4.5).

Таблица 4.4. Значения $\Lambda_3(\Theta)$

| $\Theta = x_2$ | x_3 | Ω_3 | $\Theta = x_2$ | x_3 | Ω_3 |
|----------------|-------|------------|----------------|-------|------------|
| 0 | 0 | 40 | 3 | 0 | 64 |
| | 1 | 32 | | 1 | 36 |
| | 2 | 38 | | 2 | 25 |
| | - | - | | 3 | 14 |
| | - | - | | 4 | 39 |
| 1 | 0 | 50 | 4 | 2 | 32 |
| | 1 | 22 | | 3 | 21 |
| | 2 | 28 | | 4 | 29 |
| 2 | 0 | 57 | 5 | 2 | 39 |
| | 1 | 29 | | 3 | 28 |
| | 2 | 18 | | 4 | 36 |
| | 3 | 24 | | - | - |

Третий шаг. Вычисляем для всех возможных значений Θ

$$\Lambda_2(\Theta) = \min_{x_2} \{f_2(x_2 - \Theta) + g_2(x_2 - m_2) + \Lambda_1(x_2)\} = \min \Omega_2(x_2, \Theta).$$

Результаты вычислений заносим в таблицы 4.6. и 4.7.

Таблица 4.6. Значения $\Lambda_2(\Theta)$

| x_1 | x_2 | Ω_2 | x_1 | x_2 | Ω_2 |
|-------|-------|------------|-------|-------|------------|
| 0 | 0 | 87 | 3 | 1 | 58 |
| | 1 | 76 | | 2 | 36 |
| | 2 | 71 | | 3 | 42 |
| | 3 | 66 | | - | - |
| | 4 | 72 | | - | - |
| 1 | 2 | 61 | 4 | 2 | 65 |
| | 3 | 56 | | 3 | 43 |
| | 4 | 62 | | 4 | 32 |
| | - | - | | 5 | 38 |
| 2 | 2 | 51 | 5 | 4 | 39 |
| | 3 | 46 | | 3 | 28 |
| | 4 | 52 | | 6 | 46 |

Таблица 4.5. Минимумы $\Lambda_3(\Theta)$

| $\Theta = x_2$ | Λ_3 | X_3 | x_4 |
|----------------|-------------|-------|-------|
| 0 | 32 | 1 | 1 |
| 1 | 22 | 1 | 1 |
| 2 | 18 | 2 | 1 |
| 3 | 14 | 3 | 1 |
| 4 | 21 | 3 | 1 |
| 5 | 28 | 3 | 1 |

Таблица 4.7. Минимумы $\Lambda_2(\Theta)$

| $\Theta = x_1$ | Λ_2 | X_2 | X_3 | x_4 |
|----------------|-------------|-------|-------|-------|
| 0 | 66 | 3 | 3 | 1 |
| 1 | 56 | 3 | 3 | 1 |
| 2 | 46 | 3 | 3 | 1 |
| 3 | 36 | 3 | 3 | 1 |
| 4 | 32 | 4 | 3 | 1 |
| 5 | 28 | 5 | 3 | 1 |

Таблица 4.8. Минимумы $\Lambda_1(\Theta)$

| $\Theta = x_0$ | X_1 | Ω_1 |
|----------------|-------|------------|
| 2 | 1 | 74 |
| | 2 | 46 |
| | 3 | 54 |

Четвёртый шаг. Используя начальное условие $\Theta = x_0$ и подставляя его в соотношение

$$\Lambda_1(x_0) = \min_{x_1} \{f_1(x_1 - x_0) + g_1(x_1 - m_1) + \Lambda_2(x_1)\},$$

находим $\Lambda_1(x_0) = 2$ и оптимальное значение переменной $x_1^* = 2$. Результаты расчётов приведены в табл. 4.8. Далее из табл. 4.8 по значению $\Theta = x_1^* = 2$ находим оптимальные значения всех остальных переменных $x_2^* = 3, x_3^* = 3, x_4^* = 1$. При этом минимальные суммарные затраты составят $\Lambda_1 = 46$ единиц.

Динамическое программирование больше всего подходит для задач, где переменные – дискретные величины. В то же время оно может быть приспособлено к задачам с непрерывными переменными. Для этого непрерывные переменные квантуют, причём шаг дискретизации выбирают из конкретного содержания задачи.

4.4. Задача о нахождении наиболее экономного маршрута перевозки грузов

Задача. На данной сети дорог (рис. 4.2) указаны стоимости доставки единицы груза из пункта в пункт. Найти наиболее экономный маршрут перевозки груза из пункта 1 в пункт 10.

Решение. Если затраты минимальны при перевозке единицы груза, то они будут минимальны и при перевозке любого количества груза.

Разобьём все пункты сети на группы (табл. 4.9). К i -й группе отнесём пункты, в которые можно попасть из любого пункта $(i-1)$ -й группы.

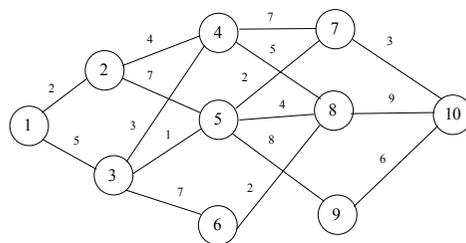


Рис 4.2. Граф сети дорог

Таблица 4.9. Группировка пунктов сети

| I | II | III | IV | V |
|---|----|-----|----|----|
| 1 | 2 | 4 | 7 | 10 |
| | 3 | 5 | 8 | |
| | | 6 | 9 | |

Формирование наиболее экономного маршрута может быть реализовано за четыре шага.

Физическая система – транспорт с грузом, перемещающийся из начального состояния C_1 (из пункта 1) в конечное C_{10} (пункт 10) и сеть дорог, где C_i – состояние системы перед i -м шагом. Естественно принять местонахождение транспорта с грузом в пункте, в котором он пребывает перед этим шагом. Управление u_i на i -м шаге состоит в выборе маршрута (i, j) , по которому следует направлять груз из данного в общем направлении на пункт 10. Состояние в конце шага определяется номером пункта, в который будет доставлен груз в результате сделанного выбора (принятого управления). Z_i (значение целевой функции на i -м шаге) – это затраты на перевозку единицы груза из данного пункта в выбранный соседний пункт. F_i определяется из формулы (*).

Первый этап: $i=N=4$ (x_i, x_{i-1} – состояния, u_i – управления).

Таблица 4.10. Результаты вычислений для 4-го шага

| x_3 | u_4 | x_4 | F_4 |
|-------|---------|----------|-------|
| C_7 | (7, 10) | C_{10} | 3* |
| C_8 | (8, 10) | C_{10} | 9 |
| C_9 | (9, 10) | C_{10} | 6 |

Второй этап: $i=3$.

Таблица 4.11. Результаты вычислений для 3-го шага

| x_2 | u_3 | x_3 | z_3 | F_4 | z_3+F_4 | F_3 |
|-------|--------|-------|-------|-------|-----------|-------|
| C_4 | (4, 7) | C_7 | 7 | 3 | 10 | 10 |
| | (4, 8) | C_8 | 5 | 9 | 14 | - |
| C_5 | (5, 7) | C_7 | 2 | 3 | 5 | 5* |
| | (5, 8) | C_8 | 4 | 9 | 13 | - |
| | (5, 9) | C_9 | 8 | 6 | 14 | - |
| C_6 | (6, 8) | C_8 | 2 | 9 | 11 | 11 |

Третий этап: $i=2$.

Таблица 4.12. Результаты вычислений для 2-го шага

| x_1 | u_2 | x_2 | z_2 | F_3 | z_2+F_3 | F_2 |
|-------|--------|-------|-------|-------|-----------|-------|
| C_2 | (2, 4) | C_4 | 4 | 10 | 14 | - |
| | (2, 5) | C_5 | 7 | 5 | 12 | 12 |
| C_3 | (3, 4) | C_4 | 3 | 10 | 13 | - |
| | (3, 5) | C_5 | 1 | 5 | 6 | 6* |
| | (3, 6) | C_6 | 7 | 11 | 18 | - |

Четвёртый этап: $i=1$.

Таблица 4.13. Результаты вычислений для 1-го шага

| x_0 | u_1 | x_1 | z_1 | F_2 | z_1+F_2 | F_1 |
|-------|--------|-------|-------|-------|-----------|-------|
| C_1 | (1, 2) | C_2 | 2 | 12 | 14 | - |
| | (1, 3) | C_3 | 5 | 6 | 11 | 11* |

Из последней таблицы видно, что условно-оптимальным управлением для этого шага является выбор маршрута (1,3), обеспечивающий минимальные суммарные затраты $F_1(x_0, u_1)=F_1(1, (1, 3))=11$ на всём пути от пункта 1 до пункта 10. Структура этого пути (наиболее экономный маршрут) выяснится в процессе безусловной оптимизации.

Для этого необходимо пройти ещё раз весь оптимизируемый процесс, начиная I и кончая IV шагом. Искомое оптимальное управление (наиболее экономный маршрут) будет составлено из найденных при условной оптимизации шаговых условно-оптимальных управлений (дорог между отдельными

пунктами сети).

Табл. 4.13. Из пункта 1 (состояние C_1) груз следует направить по дороге (1, 3) (т. к. этому шаговому управлению соответствуют минимальные затраты F_1) в пункт 3 (состояние C_3).

Табл. 4.12. Из пункта 3 груз следует направить по дороге (3, 5) в пункт 5.

Табл. 4.11. Из пункта 5 груз следует направить по дороге (5, 7) в пункт 7.

Табл. 4.10. Из пункта 7 груз следует направить по дороге (7, 10) в пункт 10.

Вывод: наиболее экономный маршрут пролегает через пункты 1, 3, 5, 7, 10, при этом транспортные расходы минимизируются и составляют 11 денежных единиц на единицу груза.

4.5. Решение задачи конвейерного типа с помощью алгоритма Джонсона

Дано: m различного типа деталей, которые необходимо обработать последовательно на двух станках (сначала на одном, затем – на другом). Имеются данные о временах обработки деталей (табл. 4.14). Последовательность обработки деталей на I и II станках одинакова.

Таблица 4.14. Продолжительность обработки деталей на станках

| Номер детали | Время обработки, усл. ед. | |
|--------------|---------------------------|------------------|
| | на I станке (A) | на II станке (B) |
| 1 | 5 | 1 |
| 2 | 2 | 6 |
| 3 | 4 | 4 |
| 4 | 3 | 1 |
| 5 | 3 | 4 |
| 6 | 0 | 2 |

Таблица 4.15. Оптимальная последовательность обработки деталей

| Номер детали | Время обработки, усл. ед. | |
|--------------|---------------------------|------------------|
| | на I станке (A) | на II станке (B) |
| 6 | 0 | 2 |
| 2 | 2 | 6 |
| 5 | 3 | 4 |
| 3 | 4 | 4 |
| 4 | 3 | 1 |
| 1 | 5 | 1 |

Требуется построить временную диаграмму обработки деталей на двух станках в заданном порядке и после упорядочения по алгоритму Джонсона.

На основании предыдущей таблицы строим временную диаграмму (рис 4.3).

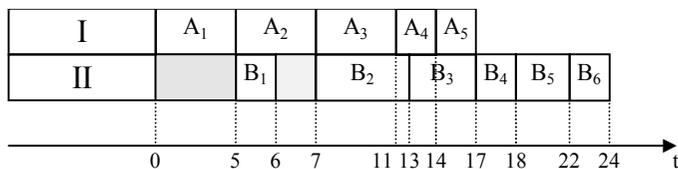


Рис 4.3. Временная диаграмма обработки деталей на двух станках

Для оптимизации цикла обработки деталей на двух станках используется алгоритм Джонсона.

1. Находятся минимальные элементы I и II столбцов табл. 4.14. Если окажется, что наименьшее число находится в I столбце, то соответствующая деталь ставится в начало очереди после ранее поставленных в начало очереди деталей (то же делаем в случае равенства наименьших значений столбцов).

2. Если же наименьшее из минимумов находится во II столбце, то соответствующая деталь ставится в конец очереди на обработку перед ранее поставленными в конец очереди деталями (табл. 4.15).

Таким образом, временная диаграмма оптимальной обработки на двух станках построена (рис 4.4).

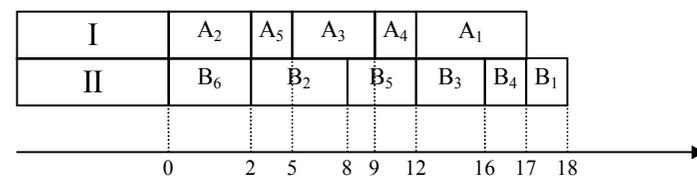


Рис 4.4. Итоговая временная диаграмма оптимальной обработки деталей на двух станках

4.6. Модель процесса эксплуатации лесного хозяйства

Содержательное описание. Леспромхоз намечает засадить лесом новый участок земли. Дерево, сваленное в начале отрезка времени (например, года k , где k – отсчитывается с момента посадки), приносит доход R_k в рублях. Предполагается, что затраты на посадку и содержание леса пренебрежимо малы по сравнению с затратами на свалку и транспортировку деревьев в начале отрезка k . Предполагается также, что все деревья должны быть срублены одновременно. Требуется определить оптимальный момент (период) k для рубки леса, при котором достигается максимальный эффект при поправочном коэффициенте α .

Случай 1. После рубки леса не будут производиться новые посадки (однократное принятие решений). Необходимо найти такое k , при котором значение интегрального эффекта (ИНЭ) является максимальным:

$$\text{ИНЭ}[k] = \alpha^{k-1} \cdot R_k \rightarrow \max_k$$

Искомое значение k определяется из итеративного соотношения:

$$\frac{R_{k-1}}{R_k} < \alpha \leq \frac{R_k}{R_{k+1}} \quad (4.5)$$

Случай 2. Рассмотрим бесконечный плановый период. Допустим, что всякий раз после рубки леса на k -м отрезке осуществляются новые лесопосадки на $(k+1)$ -м отрезке. Причём, этот процесс повторяется бесконечное число раз. Если каждый раз валка леса осуществляется в начале k -го отрезка после посадки, то ИНЭ последовательного ряда прибылей в течение бесконечного планового периода будет равен:

$$\begin{aligned} F(k) &= \text{ИНЭ}[k](1 + \alpha^k + \alpha^{2k} + \dots); \\ F(k) &= \text{ИНЭ}[k] + \alpha^k \cdot \text{ИНЭ}[k](1 + \alpha^k + \alpha^{2k} + \dots); \\ F(k) &= \text{ИНЭ}[k] + \alpha^k \cdot F(k), \end{aligned}$$

$$\text{откуда } F(k) = \frac{\text{ИНЭ}[k]}{1 - \alpha^k}.$$

Оптимальной для бесконечного планового периода является такая стратегия, при которой максимизируется величина $F(k)$:

$$F^k = \max_{k=1,2,\dots} F(k) = \max_{k=1,2,\dots} \frac{\text{ИНЭ}[k]}{1 - \alpha^k}.$$

Искомое значение k^* , при котором $F(k)$ достигает максимума, определяется из условия

$$\frac{\text{ИНЭ}[1]}{1 - \alpha^1} \leq \frac{\text{ИНЭ}[2]}{1 - \alpha^2} \leq \frac{\text{ИНЭ}[k^*]}{1 - \alpha^{k^*}} \geq \frac{\text{ИНЭ}[k+1]}{\alpha^k - 1},$$

откуда приходим к условию максимизации

$$\frac{(1 + \alpha + \dots + \alpha^{k^*-1}) R_{k^*-1}}{(1 + \alpha + \dots + \alpha^{k^*-2}) R_{k^*}} < \alpha \leq \frac{(1 + \alpha + \dots + \alpha^{k^*}) R_{k^*}}{(1 + \alpha + \dots + \alpha^{k^*-1}) R_{k^*+1}}. \quad (4.6)$$

При этом искомое значение k^* удовлетворяет неравенству $k^* \leq K$, где K – оптимальное значение для однократного принятия решений.

5. ИГРОВЫЕ МОДЕЛИ ОПЕРАЦИЙ

5.1. Модели теории игр. Основные определения и термины

В разных областях целенаправленной деятельности, например при разработке и эксплуатации АСУ, часто возникают конфликтные ситуации, характеризующиеся наличием противоположных интересов коллективов, которые пытаются достичь своих целей часто в ущерб друг другу. Размеры ущерба и его конкретное выражение могут быть самыми разными. Особое место в изучении проблем конфликта занимает выбор и сравнительный анализ возможных (допустимых) способов поведения противоборствующих сторон, что даёт основу для принятия каждой стороной разумных решений относительно своих действий. При этом стороны, принимающие решение в рассматриваемых условиях, должны учитывать не только поставленные перед собой цели, но и цели противоборствующей стороны.

Каждая сторона, участвующая в конфликте, является активной составляющей. Она формулирует свои цели, имеет средства для их достижения, разрабатывает и оценивает по принятым критериям стратегии, осуществляет рациональный (оптимальный) выбор поведения применительно к обстановке. Это означает, что стороны ведут своеобразную игру с различными противниками. Раздел исследования операций, связанный с математическим моделированием условий конфликта и поиском на этой основе оптимальных решений, называют теорией игр.

В дальнейшем термин “игра” будет употребляться только в значении “математическая модель конфликта”, а противоборствующие стороны будут называться игроками и обозначаться символами A и B . Чтобы раскрыть содержание того или иного конфликта, достаточно получить ответы на следующие вопросы: “какие стороны участвуют в конфликте?”, “какие стратегии может применять каждая сторона?”, “каковы ожидаемые результаты применения сторонами своих стратегий?”, “кто заинтересован в указанных результатах?”, “в чём выражается эта заинтересованность?”. Математическая интерпретация поставленных вопросов и возможных ответов на них позволяет строить и исследовать игровые модели операций в широком диапазоне исходных условий.

Обычно строится упрощённая ММ без приводящих факторов, называемая “игрой”. От реальной ситуации игра отличается тем, что ведётся по определённым правилам. Стороны, участвующие в игре, называются “игроками”, а результат их столкновения обозначается как “выигрыш” одного из них. Игра может быть “парной” или “множественной”. Участники множественной игры могут образовывать коалиции постоянные или временные. Под правилами игры понимается система условий, реализующая возможные ва-

рианты действий обеих сторон, объём информации у каждой стороны о поведении другой стороны, последовательность чередования действий (ходов) игроков, результат (исход) игры, к которому приводит совокупность ходов игроков.

Игра называется “игрой с нулевой суммой”, когда один игрок выигрывает то, что проигрывает другой (сумма выигрышей равна нулю). При анализе рассматривается выигрыш только одного игрока (“мы”, обозначаемого в дальнейшем как A , “противника” обозначаем как B). Всегда рассматривается A “выигрывающим”, а B всегда “проигрывает”. Ход в игре означает выбор одного из предусмотренных правилами игры вариантов. Различаем личные и случайные ходы. Личные ходы – это сознательный выбор одного из возможных вариантов в данной игровой ситуации. При случайном выборе вариантов задаётся распределение вероятностей возможных исходов. Некоторые игры могут состоять только из случайных ходов (азартные игры, например игра в “кости”). Игры могут состоять также только из личных ходов (например, шахматы) или из смешанных (личных и случайных, например, карточные игры). Стратегией игрока называется совокупность правил, определяющих однозначно выбор хода в зависимости от ситуации в игре. По числу возможных стратегий игры бывают конечные и бесконечные. В конечной игре игрок A имеет m стратегий, а игрок B использует n стратегий. Такая игра называется “игрой $m \times n$ ”. Обозначим эти стратегии соответственно $\{A_1, A_2, \dots, A_m\}$ и $\{B_1, B_2, \dots, B_n\}$. Если игра состоит из личных ходов, то выбор пары стратегий $\{A_i, B_j\}$ однозначно определяет выигрыш игрока A , обозначаемый a_{ij} . Если игра кроме личных ходов содержит ещё и случайные ходы, то при паре стратегий $\{A_i, B_j\}$ выигрыш есть случайная величина, которую можно оценить математическим ожиданием и также обозначить как a_{ij} .

Если известны все значения выигрышей при выборе стратегий игроками, то их можно представить в виде матрицы игры $\|a_{ij}\|$. На пересечении i -й строки (стратегии игрока A) с j -м столбцом (стратегией игрока B) в матрице игры находится значение выигрыша игрока A (a_{ij}). Основой игры является предположение, что противник, по меньшей мере, так же разумен, как и мы, и делает всё для того, чтобы помешать нам добиться своей цели. Фактически в ММ игры не учитывается элемент риска, просчёты и ошибки игроков. Важным ограничением ММ игры является то, что выигрыш искусственно сводится к одному единственному числу. На практике приходится учитывать несколько критериев успеха. Стратегия, оптимальная по одному критерию, необязательно будет оптимальной по другим критериям.

Для определения оптимальных стратегий поведения сторон в состязательных задачах используются:

- для игровых моделей с полной информацией – наиболее часто аппарат линейного программирования;

- при наличии неполной информации (риска) – специальные алгоритмы теории игр;

- при неопределенности поведения игроков – теория статистических решений.

Можно предложить следующую методику решения состязательных задач первого класса. Во-первых, обязательно составление матрицы выигрышей сторон при сочетании стратегий игроков.

Во-вторых, необходимо выявить основные особенности, взаимосвязи и количественные закономерности. На основании этих взаимосвязей устанавливается уравнение выигрышей конфликтующих сторон. Для каждой из конфликтующих сторон необходимо составить столько уравнений взаимосвязей, сколько имеется у нее стратегий. В этих уравнениях связей число составляющих соответствует числу стратегий, имеющихся в распоряжении у конфликтующей стороны. Обычно предполагается свойство аддитивности выигрышей. Поэтому средние выигрыши игроков представляют собой сумму эффектов.

Каждый эффект представляет собой произведение выигрыша игрока, взятый из матрицы выигрышей, на вероятность появления этой стратегии в его смешанной стратегии.

В-третьих, сводят состязательную задачу с нулевой суммой к задаче линейного программирования. В результате преобразований целевой функцией является средний выигрыш одной из конфликтующих сторон. А в качестве ограничений используются неравенства, определяющие соотношение между ценой игры и ожидаемым выигрышем одного из игроков для каждой его стратегии поведения.

В-четвертых, для облегчения решения задачи линейного программирования в ряде случаев пытаются дать геометрическое толкование решению в игровой форме, представив области ограничений и градиент изменения целевой функции.

Чтобы лучше представить методику построения моделей теории игр, обратимся к простому примеру. Пусть предприятия A и B могут выпускать одинаковую продукцию нескольких видов и должны разработать свои производственные планы с учётом имеющегося спроса на неё. Существует опасность того, что стремление каждого предприятия самостоятельно обеспечить выпуск либо всех, либо произвольно взятых видов продукции приведёт к нежелательным последствиям, а именно к перепроизводству одних и нехватке других изделий. Сбалансировать эту ситуацию поможет игровая модель, в которой множество конкурирующих сторон ограничивается указанными A и B . Их стратегии S_{1a}, S_{2a} и S_{1b}, S_{2b} соответственно заключаются в выборе различных вариантов производственного плана, а возможные результаты действий (исходы) определяются содержанием принятых планов. Так, если сторона A реализует стратегию (план) S_{1a} , а сторона B – стратегию

(план) S_{jb} , то в итоге потребители получают только ту продукцию, которая предусмотрена этими планами (исход I_{ij}). Выигрыш (прибыль) A при этом составит $PA(I_{ij})$ единиц, а выигрыш B – $PB(I_{ij})$ единиц. Следовательно, можно говорить о заинтересованности A и B в тех или иных исходах. Обычно в матрице игры сразу указывается выигрыш игрока A (a_{ij}), который эквивалентен $PA(I_{ij})$.

Если принято $PA(I_{ij}) = -PB(I_{ij}) = a_{ij}$, то возникает бескомпромиссная ситуация (всё, что выигрывает A , проигрывает B и наоборот). Если же в качестве a_{ij} рассматривается какая-то комбинация величин $PA(I_{ij})$, $PB(I_{ij})$ (сумма, среднее и т.д.), то тем самым утверждается возможность объединения усилий сторон, ведения переговоров между ними, заключения соглашений в интересах достижения и личных, и общих целей. Таким образом, главным в теории игр является принцип рационального выбора сторонами своих действий. Решающая роль здесь принадлежит информации, которой располагают стороны. Основные вопросы, определяющие содержание исследований игровых моделей, состоят в следующем: “что считается рациональным (оптимальным) решением игры?”, “существует ли в данной игре решение, которое можно назвать оптимальным?”, “как найти оптимальные решения игры?”. Отбор критериев обычно выходит за рамки ММ, хотя и оказывает определяющее влияние на характер получаемых решений. Поэтому в дальнейшем будем считать, что уже известны значения функций выигрыша или сами выигрыши конкурирующих сторон.

5.2. Принцип минимакса. Геометрическая интерпретация игровых моделей

Пусть у игроков A и B имеется соответственно m и n стратегий: (A_1, A_2, \dots, A_m) и (B_1, B_2, \dots, B_n) . Задана также матрица выигрыша $\|a_{ij}\|$, $i = \overline{1, m}$, $j = \overline{1, n}$. Такая матрица представляет собой нормальную форму игр и называется платёжной (игровой) матрицей, указывающая платежи a_{ij} (выигрыши-проигрыши) участников A и B . Сама же игра называется игрой $(m \times n)$.

Выбирая стратегию A_i , игрок A всегда должен рассчитывать, что противник B ответит той стратегией B_j , для которой его выигрыш a_{ij} будет минимальным, т. е. значение выигрыша A $\alpha_i = \min_j a_{ij}$ (минимальный элемент в i -й

строке платёжной матрицы даст j -е значение). Но у игрока A имеется m стратегий (m строк платёжной матрицы), поэтому можно остановиться на той стратегии A_i , для которой α_i максимально. Таким образом, оптимальное значение выигрыша игрока A

$$\alpha = \max_i \min_j a_{ij}. \quad (5.1)$$

Номер строки i платёжной матрицы, у которой α_i максимально, и даст оптимальную стратегию игрока A . Величина α называется нижней ценой игры (максиминный выигрыш или максимин). Стратегия игрока A , которая соответствует максимину, называется максиминной стратегией, т. е. при любом поведении игрока B игрок A гарантирует себе выигрыш не меньше α (нижняя цена игры). Это чисто перестраховочная стратегия.

Аналогичные рассуждения можно провести и за противника B . Он заинтересован, чтобы обратить выигрыш A в минимум и максимизировать свой выигрыш, т. е. $\beta_j = \max_i a_{ij}$ (максимальный элемент по j -му столбцу платёжной матрицы). Но игрок B знает, что игрок A также умён и осторожен, и позволит ему выиграть минимум из этих максимумов. Поэтому оптимальное значение выигрыша игрока B

$$\beta = \min_j \max_i a_{ij}. \quad (5.2)$$

Номер столбца j платёжной матрицы, у которого β_j минимально, и даст оптимальную стратегию игрока B . Величина β называется верхней ценой игры (минимаксный выигрыш или минимакс). Стратегия игрока B , которая соответствует минимаксу, называется минимаксной. Игрок B гарантирует себе следующее: что бы ни предпринимал игрок A против него, он во всяком случае проиграет не больше, чем β .

Принцип осторожности обоих игроков, диктующий им выбор соответствующих стратегий, называется “принципом минимакса”. Если $\alpha = \beta$, то говорят, что игра имеет **чистую цену игры** $v = \alpha = \beta$. В этом случае минимаксные стратегии становятся устойчивыми. Элемент в платёжной матрице $\|a_{ij}\|$, который имеет одновременно минимум по строке и максимум по столбцу, называют седловой точкой. Про такую игру говорят, что она имеет седловую точку. Седловая точка соответствует паре оптимальных стратегий обоих игроков (A_i, B_j) . Эти стратегии и будут решением игры.

Если один из игроков (A) придерживается своей оптимальной стратегии, а другой игрок (B) будет отклоняться от своей оптимальной стратегии, то последний от этого ещё больше проиграет. Итак, для каждой игры с седловой точкой существует решение, определяющее пару оптимальных стратегий обоих сторон, отличающихся следующими свойствами:

- 1) средний выигрыш равен чистой цене игры v , являющейся её верхней (β) и нижней (α) ценой;
- 2) если одна из сторон придерживается своей оптимальной стратегии, а другая отклоняется от своей, то от этого отклоняющаяся сторона может только потерять, а не увеличить свой выигрыш.

Ставится вопрос: нельзя ли гарантировать себе выигрыш больше α , если

применять не одну стратегию A_i , а чередовать случайным образом несколько стратегий? Каждая чистая стратегия есть частный случай смешанной, в которой все стратегии, кроме одной, применяются с нулевыми частотами, а данная – с частотой, равной единице. Для смешанных стратегий игрока A , например (A_1, A_2, A_3) с частотами $(p_1 + p_2 + p_3 = 1)$, применяется следующая форма записи:

$$S_A = \begin{pmatrix} A_1 & A_2 & A_3 \\ p_1 & p_2 & p_3 \end{pmatrix}.$$

Аналогично смешанные стратегии игрока B с частотами $(q_1 + q_2 + q_3 = 1)$ обозначаются в виде:

$$S_B = \begin{pmatrix} B_1 & B_2 & B_3 \\ q_1 & q_2 & q_3 \end{pmatrix}.$$

Если найдено оптимальное решение игры, состоящее из двух смешанных стратегий (S_A^*, S_B^*) , то стратегии, входящие в оптимальные стратегии игроков, называются “полезными”. Если один из игроков придерживается своей оптимальной смешанной стратегии S_A^* , (S_B^*) , то выигрыш его остаётся неизменным и равным цене игры v независимо от того, что делает другой игрок. Иногда задачу анализа игры можно упростить, если удаётся построить её геометрическую интерпретацию.

Для примера рассмотрим игру (2×2) с матрицей игры, равной

| | | |
|-------|----------|----------|
| | B_1 | B_2 |
| A_1 | a_{11} | a_{12} |
| A_2 | a_{21} | a_{22} |

Нужно найти $S_A^* = \begin{pmatrix} A_1 & A_2 \\ p_1 & p_2 \end{pmatrix}$, оптимальную стратегию игрока A , т. е. определить вероятности p_1 и p_2 чередования стратегий A_1 и A_2 . Составим уравнение для цены игры:

$$a_{11}p_1 + a_{12}p_2 = v; \quad a_{21}p_1 + a_{22}p_2 = v; \quad p_1 + p_2 = 1; \quad p_2 = 1 - p_1.$$

Подставим p_1 в данную систему и приравняем первые два уравнения:

$$a_{11}p_1 + a_{12}(1 - p_1) = a_{21}p_1 + a_{22}(1 - p_1).$$

После преобразований находим $p_1 = \frac{a_{22} - a_{21}}{a_{11} + a_{12} - a_{12} - a_{21}}$.

Цену игры v и p_2 находим, подставляя значение p_1 в предыдущие уравнения.

Решению игры (2×2) можно дать простую геометрическую интерпретацию. Поскольку $0 \leq a_{ij} \leq 1$ и вероятности $0 \leq p_k \leq 1$, то в системе координат a_{ij} (ординаты) и p_2 (абсциссы) построим квадрат со сторонами, равными 1. Вертикальные стороны этого квадрата соответствуют $p_2 = 0$ (левая) и $p_2 = 1$ (правая). На них будем откладывать значения выигрыша из платёжной матрицы $\|a_{ij}\|$. В результате левая сторона квадрата будет соответствовать стратегии A_1 , а правая сторона квадрата определит стратегию A_2 игрока A . На этих сторонах отложим значения выигрышей по первой строке платёжной матрицы (a_{11} и a_{12}) и по второй строке платёжной матрицы (a_{21} и a_{22}). Соединим линиями точки, находящиеся на левой и правой сторонах квадрата соответственно столбцам. Теперь стратегией B_1 игрока B (соответственно первому столбцу платёжной матрицы) будет линия, соединяющая a_{11} с a_{21} . Аналогично стратегией B_2 игрока B (соответственно второму столбцу платёжной матрицы) будет линия, соединяющая a_{12} с a_{22} . На рис 5.1 представлен пример, когда платёжная матрица $\|a_{ij}\|$ имеет седловую точку M . Она получилась на пересечении стратегий игрока B . Чистая цена игры v определена ординатой точки M . Нижняя граница выигрыша игрока A определяется линией (a_{11}, M, a_{22}) , верхняя цена выигрыша игрока B – линией (a_{12}, M, a_{21}) . Точка M , в которой максимум равен минимуму, и определяет решение игры v . Для её достижения игрок A должен чередовать стратегию A_1 с вероятностью p_1^* со стратегией A_2 с вероятностью p_2^* . Значение абсциссы точки M определяет величину вероятности p_2^* .

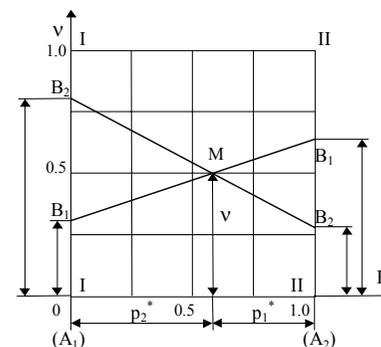


Рис 5.1. Матрица игры имеет седловую точку

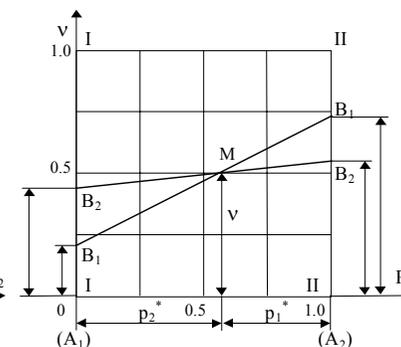


Рис 5.2. Одна из стратегий заведомо невыгодна

Возможен другой случай, когда матрица игры имеет седловую точку, а стратегия A_1 для игрока A заведомо невыгодна, так как при любой чистой стратегии даёт меньший выигрыш, чем при A_2 . Этот случай представлен на рис. 5.2. Игрок A должен применять только стратегию A_2 . Чистая цена игры

такую стратегию $S_B=(y_1, y_2)$, которая обеспечивала бы ей с вероятностью не больше v максимальный захват рынка сбыта у фирмы-конкурента. Здесь под x_i и y_j понимают вероятности стратегий соответственно фирм A и B .

Соотношения для вероятностной стратегий фирм A и B имеют вид

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 = 1; \quad y_1 + y_2 = 1.$$

Условно будем считать, что a_{ij} представляют собой вероятности захвата рынка. Тогда по формуле полной вероятности независимых событий составим два уравнения для фирмы A (на каждую из двух стратегий фирмы B):

$$\begin{aligned} a_{11}x_1 + a_{21}x_2 + a_{31}x_3 + a_{41}x_4 + a_{51}x_5 + a_{61}x_6 &\geq v; \\ a_{12}x_1 + a_{22}x_2 + a_{32}x_3 + a_{42}x_4 + a_{52}x_5 + a_{62}x_6 &\geq v; \\ 0 &\leq x_i \leq 1. \end{aligned}$$

Аналогично составим уравнения для фирмы B (на каждую из шести стратегий фирмы A):

$$\begin{aligned} a_{11}y_1 + a_{12}y_2 &\leq v; & a_{21}y_1 + a_{22}y_2 &\leq v; \\ a_{31}y_1 + a_{32}y_2 &\leq v; & a_{41}y_1 + a_{42}y_2 &\leq v; \\ a_{51}y_1 + a_{52}y_2 &\leq v; & a_{61}y_1 + a_{62}y_2 &\leq v; \\ 0 &\leq y_j \leq 1. \end{aligned} \quad (5.5)$$

Подставив $y_2=1-y_1$ во все неравенства (5.5) и рассматривая их как уравнения, получим систему уравнений

$$\begin{aligned} v &= a_{12} + (a_{11} - a_{12})y_1; & v &= a_{22} + (a_{21} - a_{22})y_1; \\ v &= a_{32} + (a_{31} - a_{32})y_1; & v &= a_{42} + (a_{41} - a_{42})y_1; \\ v &= a_{52} + (a_{51} - a_{52})y_1; & v &= a_{62} + (a_{61} - a_{62})y_1; \\ 0 &\leq y_1 \leq 1. \end{aligned} \quad (5.6)$$

Для примера пусть платёжная матрица $\|a_{ij}\|$ имеет вид:

| | | |
|-------|---------------|---------------|
| A_1 | $a_{11}=0,54$ | $a_{12}=0,75$ |
| A_2 | $a_{21}=0,64$ | $a_{22}=0,36$ |
| A_3 | $a_{31}=0,19$ | $a_{32}=0,91$ |
| A_4 | $a_{41}=0,56$ | $a_{42}=0,60$ |
| A_5 | $a_{51}=0,37$ | $a_{52}=0,85$ |
| A_6 | $a_{61}=0,46$ | $a_{62}=0,76$ |

Тогда система уравнений (5.6) примет вид

$$\begin{aligned} v &= 0,75 - 0,24y_1; \quad (1) & v &= 0,91 - 0,72y_1; \quad (3) & v &= 0,85 - 0,48y_1; \quad (5) \\ v &= 0,36 + 0,28y_1; \quad (2) & v &= 0,60 - 0,02y_1; \quad (4) & v &= 0,76 - 0,30y_1. \quad (6) \end{aligned}$$

Начертим эти шесть прямых в системе координат v 0 y_1 (рис. 5.5).

Каждая точка заштрихованной области определяет возможное решение,

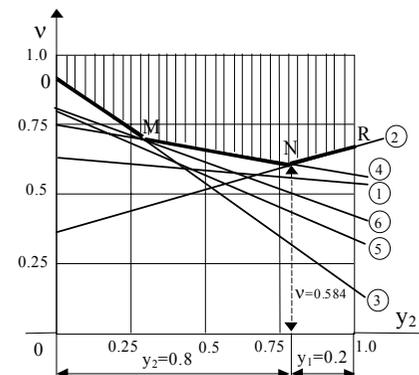


Рис 5.5. Геометрическая интерпретация стратегий фирм A и B в борьбе за рынок сбыта

ибо она удовлетворяет всем рассмотренным неравенствам для фирмы B . Из рис. 5.5 ясно, что наименьшая величина v находится на пересечении прямых (2) и (4). Приравняв эти уравнения между собой, получим:

$$\begin{aligned} 0,36 + 0,28y_1 &= 0,60 - 0,02y_1; \\ y_2 &= 1 - y_1, \end{aligned}$$

откуда находим, что $y_2=0,8$ и $y_1=0,2$. Таким образом, выбирая стратегию B_2 с вероятностью $y_2=0,8$ и стратегию B_1 с вероятностью $y_1=0,2$, фирма B имеет мини-

мальные потери от конкурентной борьбы с более мощной фирмой A . В результате рынок сбыта между фирмами будет поделен следующим образом: фирма A захватит $v=0,584$ рынка сбыта, а фирме B останется доля $1-v=0,416$ рынка сбыта. Для нахождения вероятностей стратегий фирмы A поступим следующим образом. Положим $x_1=x_3=x_4=x_5=0$. И, подставив значение $v=0,584$ в уравнения стратегий фирмы A , находим: $x_2=1/15$, а $x_4=14/15$. В итоге, если фирма A выполняет стратегию A_2 с вероятностью $1/15$ и стратегию A_4 с вероятностью $14/15$, то доля захвата рынка сбыта у фирмы A будет максимальной, и она составит $0,584$, несмотря на поведение фирмы B . Из рис. 5.5 видно, что если обе фирмы отклонятся от своих стратегий, то они теряют долю захвата рынка сбыта.

5.4. Использование теории матричных игр при выборе типов машин для строительства

Задание. Решить задачу выбора типов машин для производства строительного-монтажных работ с использованием математического аппарата теории игр.

Постановка задачи. Имеется выбор из нескольких типов машин, технологически пригодных для выполнения данного комплекса механизированных работ. Требуется однозначно определить, какие типы машин из этого набора являются для строительной организации более выгодными с точки зрения принятого критерия оценки. В качестве такого критерия могут быть приняты, например, приведённые затраты на выполнение единицы работ, себестоимость работ и др.

По условию задачи требуется привести рассматриваемые типы машин в определённую систему, т. е. произвести их упорядочение.

Упорядочение типов машин в данном случае означает расположение их в определённой последовательности, в порядке предпочтительности одной машины относительно другой, т. е. так, чтобы первый тип был бы для строительной организации выгоднее, чем второй, а второй – предпочтительнее, чем третий, и т. д.

Задача решается в условиях, когда известна номенклатура работ на планируемый период, но неизвестно, в каком пропорциональном отношении встретятся механизированные работы различных видов.

Условия задачи заданы платёжной матрицей (табл. 5.1).

Таблица 5.1. Платёжная матрица игры

| X \ Y | | Вид работ | | | | |
|-----------|-------|-----------|-------|-------|-------|-------|
| | | y_1 | y_2 | y_3 | y_4 | y_5 |
| Тип машин | x_1 | 0,80 | 0,92 | 0,95 | 0,78 | 0,70 |
| | x_2 | 0,75 | 0,90 | 0,88 | 0,76 | 0,84 |
| | x_3 | 0,95 | 0,68 | 0,70 | 0,91 | 0,60 |
| | x_4 | 0,70 | 0,77 | 0,80 | 0,65 | 0,95 |

В табл. 5.1 x_1, x_2, x_3, x_4 – типы машин, которыми может быть укомплектована строительная организация, а y_1, y_2, y_3, y_4, y_5 – виды строительно-монтажных работ, пропорциональное соотношение между которыми неизвестно.

В клетках, стоящих на пересечении строк и столбцов платёжной матрицы, приведены показатели единичной стоимости выполнения работ машинами соответствующих типов (в условных денежных единицах).

Проверим игру на наличие седловой точки.

Противник при любой нашей стратегии (т. е. для любого вида техники) пытается выбрать такой вид работ, для которого себестоимость будет максимальной:

$$\beta = \min_i \max_j a_{ij} = 0,91 \text{ – для нас.}$$

Из стратегий мы выбираем ту, которая минимизирует себестоимость (как результат данного подхода противника):

$$\alpha = \max_j \min_i a_{ij} = 0,75, \alpha \neq \beta.$$

Решение игры определяем в смешанных стратегиях. Цена игры v заключена между нижней α и верхней β ценами, т.е. $0,75 \leq v \leq 0,91$. Составим задачу линейного программирования, условия которой выражаются следующей линейной системой:

$$\begin{cases} \frac{p_1}{v} + \frac{p_2}{v} + \frac{p_3}{v} + \frac{p_4}{v} = \frac{1}{v} \rightarrow \max; \\ 0,80 \frac{p_1}{v} + 0,75 \frac{p_2}{v} + 0,95 \frac{p_3}{v} + 0,70 \frac{p_4}{v} \leq 1; \\ 0,92 \frac{p_1}{v} + 0,90 \frac{p_2}{v} + 0,68 \frac{p_3}{v} + 0,77 \frac{p_4}{v} \leq 1; \\ 0,95 \frac{p_1}{v} + 0,88 \frac{p_2}{v} + 0,70 \frac{p_3}{v} + 0,80 \frac{p_4}{v} \leq 1; \\ 0,78 \frac{p_1}{v} + 0,76 \frac{p_2}{v} + 0,91 \frac{p_3}{v} + 0,65 \frac{p_4}{v} \leq 1; \\ 0,70 \frac{p_1}{v} + 0,84 \frac{p_2}{v} + 0,60 \frac{p_3}{v} + 0,95 \frac{p_4}{v} \leq 1, \end{cases}$$

где p_1, p_2, p_3, p_4 – искомые пропорции по типам машин; v – гарантированная величина средней себестоимости работ, выше которой данный показатель не поднимется при любом, даже самом неблагоприятном соотношении работ.

Решая задачу симплекс-методом, находим оптимальный план ($\frac{p_1}{v} = 0,203, \frac{p_2}{v} = 0, \frac{p_3}{v} = 0,405, \frac{p_4}{v} = 0,658$) и оптимальное значение целевой функции ($1/v = 1,265$). Отсюда $v = 0,79; p_1 = 0,16; p_2 = 0; p_3 = 0,32; p_4 = 0,52$. Следовательно, если пропорции между различными типами машин в составе парка будут соответствовать найденным значениям p_1, p_2, p_3, p_4 , то средняя себестоимость выполнения единицы строительно-монтажных работ в целом по парку не превысит 0,79 (ден. ед.) даже в самом худшем случае. Наиболее предпочтительным является тип машин x_4 , затем – x_3 , затем – x_1 , а использование типа машин x_2 экономически нецелесообразно.

ЗАКЛЮЧЕНИЕ

ИСО как наука располагает разнообразными средствами моделирования целенаправленной деятельности и современной методологией поиска наилучших решений. Студентам, желающим глубже познакомиться с практикой использования ММ при анализе сложных технических систем, авторы рекомендуют обширный список литературы. Кроме того, во второй и в третьей частях пособия будут даны методики решения задач производства с помощью соответственно вероятностных и имитационных методов математического моделирования.

ОГЛАВЛЕНИЕ

СПИСОК ЛИТЕРАТУРЫ

1. Вагнер Г. Основы исследования операций. – М.: Советское радио, 1972. Т. 1 – 329 с., Т. 2 – 480 с., Т. 3 – 561 с.
2. Вентцель Е. С. Исследование операций. – М.: Советское радио, 1979. – 575 с.
3. Зайченко Ю. П. Исследование операций. – Киев: Вища школа, 1975. – 590 с.
4. Казанцев В. П., Ярошевич В. П. Применение математических методов в инженерных и экономических расчётах: Уч. пособие. – Гомель: БелИИЖТ, 1971. – 56 с.
5. Максимей И. В. Модели задач исследования операций: Учеб. пособие. – Гомель: ГГУ, 1984. – 76 с.
6. Бахтин А. Е. Дискретные задачи производственно-транспортного типа. – Новосибирск: Наука, 1978. – 160 с.
7. Дегтярёв Ю. И. Исследование операций: Учебник для вузов спец. АСУ. – М.: Высшая школа, 1986. – 320 с.
8. Кофман А. Займёмся исследованием операций. – М.: Мир, 1966. – 278 с.
9. Ярошевич В. П. Применение линейного программирования для совершенствования эксплуатационной работы транспорта: Уч. пособие. — Гомель: БелИИЖТ, 1975. – 34 с.

| | |
|---|----|
| ВВЕДЕНИЕ..... | 3 |
| 1. ОСНОВНЫЕ ПОНЯТИЯ ТЕОРИИ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ | 4 |
| 1.1. Кибернетический подход к исследованию технических систем | 4 |
| 1.2. Способы формализации технических систем | 5 |
| 1.3. Отличительные особенности исследования операций | 7 |
| 2. МОДЕЛИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ | 8 |
| 2.1. Каноническая форма. Прямая и двойственная формулировка линейных моделей..... | 8 |
| 2.2. Примеры составления линейных моделей..... | 10 |
| 2.3. Симплекс-метод решения задачи линейного программирования | 12 |
| 2.3.1. Общие определения | 12 |
| 2.3.2. Метод исключений Жордана-Гаусса | 13 |
| 2.3.3. Алгебра симплекс-метода..... | 14 |
| 2.3.4. Вычислительная схема симплекс-метода для задачи в базисной форме..... | 16 |
| 2.3.5. Алгоритм симплекс-метода | 16 |
| 2.3.6. Построение начального опорного плана..... | 17 |
| 2.4. Анализ линейных моделей на чувствительность | 18 |
| 2.4.1. Варьирование ресурсов..... | 18 |
| 2.4.2. Варьирование целевой функцией | 19 |
| 2.5. Примеры применения моделей линейного программирования | 19 |
| 2.5.1. Составление линейных моделей | 19 |
| 2.5.2. Решение задачи линейного программирования симплекс-методом | 20 |
| 2.5.3. Пример решения транспортной задачи методом потенциалов .. | 21 |
| 2.5.4. Пример графического решения задач линейного программирования | 24 |
| 2.5.5. Примеры анализа линейных моделей на чувствительность..... | 25 |
| 3. СЕТЕВЫЕ И ГРАФОВЫЕ МОДЕЛИ | 28 |
| 3.1. Модели на графах..... | 28 |
| 3.1.1. Задача о наибольшем потоке в сети | 28 |
| 3.1.2. Нахождение кратчайшего пути в транспортной сети | 31 |
| 3.1.3. Построение графа наименьшей длины | 33 |
| 3.1.4. Задача об оптимальном назначении | 33 |
| 3.2. Модели сетевого планирования | 34 |
| 3.2.1. Технология построения сетевых графиков | 34 |
| 3.2.2. Расчёт параметров сетевого графика при фиксированных временах выполнения работ..... | 35 |

| | |
|---|----|
| 3.2.3. Расчёт параметров сетевого графика при случайных значениях времён свершения работ | 37 |
| 3.2.4. Оптимизация плана комплекса работ..... | 37 |
| 3.3. Примеры применения графовых моделей и моделей сетевого планирования..... | 38 |
| 3.3.1. Задача оптимизации документооборота | 38 |
| 3.3.2. Моделирование станционной технологии с помощью моделей сетевого планирования | 39 |
| 4. МОДЕЛИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ..... | 41 |
| 4.1. Идея метода динамического программирования..... | 41 |
| 4.2. Задача о выборе маршрута движения транспортного средства..... | 42 |
| 4.3. Задача об использовании рабочей силы на предприятии..... | 43 |
| 4.4. Задача о нахождении наиболее экономного маршрута перевозки грузов | 46 |
| 4.5. Решение задачи конвейерного типа с помощью алгоритма Джонсона..... | 47 |
| 4.6. Модель процесса эксплуатации лесного хозяйства | 47 |
| 5. ИГРОВЫЕ МОДЕЛИ ОПЕРАЦИЙ..... | 48 |
| 5.1. Модели теории игр. Основные определения и термины..... | 48 |
| 5.2. Принцип минимакса. Геометрическая интерпретация игровых моделей | 50 |
| 5.3. Моделирование конкурентной борьбы двух группировок фирм за рынок сбыта технической продукции | 52 |
| 5.4. Использование теории матричных игр при выборе типов машин для строительства | 53 |
| ЗАКЛЮЧЕНИЕ | 54 |
| СПИСОК ЛИТЕРАТУРЫ..... | 55 |