Е. В. Алексеева

Построение математических моделей целочисленного линейного программирования. Примеры и задачи

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет информационных технологий

Е. В. Алексеева

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ЦЕЛОЧИСЛЕННОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ. ПРИМЕРЫ И ЗАДАЧИ

Учебное пособие

Новосибирск
2012

УДК 519.8(075.8)

ББК В183я73-1

А 471

Алексеева Е. В. Построение математических моделей целочисленного линейного программирования. Примеры и задачи: Учеб. пособие / Новосиб. гос. ун-т. Новосибирск, 2012. 131 с.

ISBN

Пособие предназначено для студентов и магистрантов Новосибирского государственного университета, изучающих дисциплины «Теория принятия решений» и «Исследование операций». Материал, содержащийся в пособии, является частью основных лекционных курсов и семинарских занятий по этим дисциплинам.

Рецензент

д-р физ.-мат. наук, проф. Ю. А. Кочетов

ISBN

Оглавление

1. Введение

Ежедневно люди принимают тысячи решений, отвечая на различные вопросы, начиная от простых «Где пообедать?» до сложных «В каких городах разместить филиалы компании?» Часто процесс принятия решения можно описать аналитически. Предположим, Вы хотите оценить время, затрачиваемое на дорогу от работы до дома, для этого нужно узнать расстояние и разделить его на среднюю скорость передвижения. Получаем *количественную модель*:

$$t = \frac{S}{v},$$

где S — расстояние, v — средняя скорость, t — время в пути. Эта модель полезна, но, как и любая модель, обладает существенным недостатком: упрощает и идеализирует реальность. Например, в модели не учитывается, что, возвращаясь с работы домой, Вы делаете остановки, чтобы зайти в магазины. Это можно учесть, добавив слагаемое:

$$t = \frac{S}{v} + nR,$$

где n — планируемое число остановок, R — среднее время на остановку. В данной модели имеются постоянные величины: расстояние между домом и офисом; и переменные величины, значениями которых можно управлять: скорость передвижения, число остановок и время посещения магазина. С помощью управляемых переменных можно сократить затрачиваемое время на дорогу. Модели, имеющие переменные величины, при изменении значений которых можно моделировать и получать разные решения, называют *моделями принятия решений*, а сами переменные называют *переменными принятия решений*. Цель таких моделей не просто вычислить значение какой-то переменной, а найти минимум или максимум какой-либо функции от переменных принятия решений, например, минимизировать потраченное время, максимизировать прибыль и т. д.

Рассмотрим следующий пример. Фирма производит шесть типов стульев: *капитан*, *помощник*, *маркиза*, *испанский*, *венский* и *офисный*. Для производства стульев необходимы универсальные детали: длинные и короткие болты, тяжелые и легкие сиденья, длинные и короткие ножки, перекладины, гайки, роллеры, каркас и крепления. В табл. 1 приводятся данные о стоимости стульев, потребности в деталях для каждого типа стульев и их наличие на складе. Благодаря универсальности деталей производитель может быстро реагировать на изменения в спросе. Поступил заказ изготовить 40 стульев каждого типа из имеющихся на складе деталей.

Таблица 1

Исходные данные задачи

Тип стула	*Офис.*	*Пом.*	*Кап.*	*Марк.*	*Исп.*	*Вен.*	
Цена	36	40	45	38	35	25	
Требуемое количество деталей							Всего на складе
длин. болты	8	0	12	0	8	4	1280
корот. болты	4	12	0	12	4	8	1900
тяж. сиденья	4	4	4	4	4	4	1090
лег. сиденья	1	0	0	0	1	1	190
длин. ножки	0	1	1	1	0	0	170
корот. ножки	6	0	4	0	5	0	1000
перекладины	0	4	0	5	0	6	1000
гайки	1	0	0	0	0	0	110
роллеры	0	1	0	0	0	0	72
каркас	0	0	1	1	0	0	93
крепления	0	0	0	0	1	1	85

Суммарный доход от продажи стульев определяется выражением:

$$p = \sum_{i=1}^{6} 40c_i,$$

где i — тип стула, c_i — стоимость одного стула i-го типа. Результаты расчетов для склада приводятся в табл. 2. Согласно этому плану доход составляет 8760 у.е., причем длинные болты израсходованы полностью. Можно ли увеличить доход фирмы, если количество стульев в заказе будет другим? Чтобы ответить на этот вопрос, проанализируем решение. Заметим, что для стула «капитан» длинных болтов требуется больше, чем для других стульев.

Таблица 2

Первый план производства

Тип стула	*Офис.*	*Пом.*	*Кап.*	*Марк.*	*Исп.*	*Вен.*
Количество	40	40	40	40	40	40
Суммарный доход	8760					
Всего использовано		Остаток				
длин. болты	1280	0				
корот. болты	1600	300				
тяж. сиденья	960	130				
лег. сиденья	120	70				
длин. ножки	120	50				
корот. ножки	600	400				
перекладины	600	400				
гайки	40	70				
роллеры	40	32				
каркас	80	13				
крепления	80	5				

Сократим производство стульев «капитан» на 2 единицы, а на вырученные 24 длинных болта произведем 3 «офисных» стула. При этом фирма потеряет от непроизводства «капитана» 90 у.е., но заработает 108 у.е., если продаст дополнительно три «офисных» стула. Таким образом, суммарный доход увеличится на $18 = (43 \cdot 36 + 40 \cdot 40 + 38 \cdot 45 + 40 \cdot 38 + 40 \cdot 35 + 40 \cdot 25 - 8760)$ у.е.

Можно ли еще увеличить доход? Проведем те же рассуждения. Для производства «испанского» стула необходимо 8 длинных болтов, а для «венского» в 2 раза меньше. Значит, если вместо одного «испанского» стула будем производить два «венских» стула, то доход составит не 35 у.е., а 50 у.е. Следовательно, если фирма будет производить вместо 40 «испанских» стульев 80 «венских» стульев, то ее суммарный доход увеличится на $600 = (40 \cdot 36 + 40 \cdot 40 + 40 \cdot 45 + 40 \cdot 38 + 0 \cdot 35 + 120 \cdot 25 - 8760)$ у.е. Результаты расчетов по этому плану приводятся в табл. 3. Однако этот план не может быть реализован из-за недостатка деталей на складе, о чем свидетельствуют отрицательные значения в столбце *остаток.*

Таблица 3

Второй план производства

Тип стула	*Офис.*	*Пом.*	*Кап.*	*Марк.*	*Исп.*	*Вен.*
Количество	40	40	40	40	0	120
Суммарный доход	9360					
Всего использовано		Остаток				
длин. болты	1280	0				
корот. болты	2320	-180				
тяж. сиденья	1200	-30				
лег. сиденья	220	30				
длин. ножки	80	50				
корот. ножки	600	600				
перекладины	1080	-80				
гайки	100	70				
роллеры	40	32				
каркас	40	13				
крепления	120	-35				

В этой задаче мы использовали количественную модель, отражающую взаимосвязь между количеством производимых стульев, количеством деталей и суммарным доходом. Однако если фирме нужно ответить на вопрос о том, сколько произвести стульев, чтобы получить максимальный доход, то количественная модель в этом не поможет, и нужно сформулировать модель принятия решений. Далее будут описаны способы построения различных моделей принятия решений, разобрано, насколько хорошей является сформулированная математическая модель и как ее улучшить.

Модели принятия решений, в которых ищется максимум (минимум) некоторой функции от переменных принятия решений при некоторых ограничениях называют также *оптимизационными моделями*. Под *ограничениями* понимают условия, которые препятствуют получению максимального (минимального) значения целевой функции, т. е. не дают, например, сократить затраты до 0 или достигнуть бесконечно большого дохода. В задаче про стулья в качестве ограничений были условия на наличие деталей на складе. В других задачах ограничения могут возникать в результате ограниченного бюджета, ограниченного времени выполнения проекта, ограниченных производственных мощностей, вместимости складов, а также из-за различных требований к производственному процессу, допустимым объемам затрат и др.

Оптимизационные модели активно стали использоваться в годы Второй мировой войны при решении военных задач. Со временем сложились научные дисциплины *исследование операций* (от англ. operations research) и *теория принятия решений* (от англ. management science), которые занимаются оптимизационными задачами, возникающими в экономике, производстве, политике и других областях человеческой деятельности. Под исследованием операций понимают применение математических и количественных методов для обоснования решений во всех областях целенаправленной человеческой деятельности [2]. Под теорией принятия решений понимают особый вид человеческой деятельности, направленный на выбор наилучшего варианта действий [11]. Четкое разделение между этими дисциплинами трудно провести, но можно сказать, что дисциплина «Теория принятия решений» оказывается шире, чем «Исследование операций», и использует ее как инструмент, когда реальную задачу удается формализовать в виде математической модели и найти наилучший вариант решения путем математических расчетов. Однако такая формализация часто оказывается труднореализуемой из-за нескольких критериев оптимизации, слабой структурированности задачи, наличия многих лиц, принимающих решения или других причин. В таких ситуациях методы исследования операций носят вспомогательный характер и выступают как инструмент теории принятия решений.

Для большинства человеческих решений нельзя точно рассчитать и оценить их последствия. Принимая решение при выборе того или иного варианта, человек интуитивно может учесть гораздо больше нюансов, чем машина. Представьте ситуацию, когда семья из нескольких человек разных поколений собирается приобрести дом. Они рассматривают несколько предложений. Каждый член семьи оценивает варианты домов исходя из собственных критериев. Старшие члены семьи считают, что главное — это развитая медицинская инфраструктура. Среднее поколение считает, что важнее всего близкое расположение офисов, в которые им приходится добираться каждое утро. Для младшего поколения (с точки зрения родителей) важнее всего наличие хорошей школы. Существуют еще и другие критерии у каждого из членов семьи: удобная парковка около дома, экологическая обстановка в районе, наличие магазинов и т.д. В подобных случаях использовать только математический аппарат для принятия решения почти невозможно, поскольку необходимо использовать уникальные умения человека соизмерять противоречивые оценки, отбрасывать второстепенное, оценивать обстановку и учитывать перспективу принятого решения.

Наиболее удобным и распространенным математическим инструментом при моделировании и решении оптимизационных задач является *линейное программирование* — специальный класс оптимизационных задач, в котором все отношения между переменными выражаются линейными функциями, а переменные принимают действительные значения. Преимущество этого класса в том, что разработаны универсальные алгоритмы для решения таких задач большой размерности. Впервые задача линейного программирования в России была сформулирована в 1939 г. Л. В. Канторовичем, который применил математическую модель этой задачи в экономике и разработал метод решения. В 1975 г. Л. В. Канторович получил Нобелевскую премию за достижения в этой области [3]. В 1947 г. американский ученый Д. Данциг разработал алгоритм решения этой задачи. С этого момента линейное программирование стало важным инструментом в исследовании операций.

Часто из условия задачи следует, что значения некоторых переменных принятия решений принадлежат множеству целых чисел. Например, если переменная отвечает за размер детали, то она может принимать значения только из заданного множества возможных размеров деталей. Такого рода задачи принадлежат к классу задач *целочисленного линейного программирования* или *смешанного целочисленного линейного программирования*. Они отличаются от задач линейного программирования высокой сложностью и особой практической значимостью. Задачам из этих классов в данном пособии уделяется особое внимание.

Рис. 1. Общая схема исследования задачи

Прежде чем переходить к построению математической модели, необходимо пройти несколько этапов. На рис. 1 приводится общая схема исследования проблемы. Она решается в интересах так называемого лица, принимающего решения (ЛПР). ЛПР формулирует проблему, участвует в построении модели, анализирует полученное решение, а затем принимает или отвергает его. Построение математической модели и решение задачи лучше доверить специалистам по исследованию операций.

Исследование задачи начинается с появления некоторой проблемы. Большинство оптимизационных задач возникает из практических приложений.
В этом случае рекомендуется изучить процесс, в результате которого возникла задача. Может оказаться, что задача не настолько сложна, чтобы для нее использовать математический аппарат. Возможно, чтобы решить поставленную задачу, необходимо найти решение какой-то другой подзадачи, в результате которой возникла эта проблема.

Далее необходимо сформулировать задачу: определить ее цель, сформулировать условия, которые влияют на достижение цели. Выяснить, какие упрощения по сравнению с реальностью могут быть допущены в модели. Реальность сложная и многогранная, поэтому учесть все нюансы сложной прикладной задачи в модели не удастся. В связи с этим на этапе уяснения и формулировки задачи необходимо провести отбор условий, которые сохранят адекватность модели и не перегрузят ее несущественными деталями.

Следующий этап состоит в том, чтобы собрать все необходимые численные данные. Понять, какие параметры измеряют и влияют на цель задачи, разбить их на управляемые параметры (переменные) и неуправляемые параметры (константы), ввести переменные, задать целевую функцию. Возможно, с первого раза не удастся наилучшим образом задать переменные. Кроме того, необходимо решить вопрос о *размерности модели*. Она определяется количеством переменных и ограничений. Если число управляемых параметров увеличить, то с помощью модели можно более точно отразить реальное событие, но будет трудно выявить основные свойства модели. В такой ситуации задача становится необозримой и может не иметь решения. Поэтому число переменных стараются уменьшить, оставляя главные. С другой стороны, уменьшая число переменных, можно опустить существенные переменные, и модель становится неадекватной. Большое количество ограничений тоже не всегда является недостатком. В следующих разделах будут рассмотрены разные модели одной и той же задачи, и читатель сможет сравнить качество каждой из этих моделей.

Выполнение только первых трех этапов из общей схемы может оказаться полезным по нескольким причинам. Во-первых, при исследовании проблемы на этапе количественного анализа нужно собрать, проверить и структурировать исходные данные задачи. Например, подсчитать имеющееся количество деталей на складе, установить, какие составные части используются в производстве, а какие, возможно, уже устарели, и их можно исключить из рассмотрения. Во-вторых, определить, какие параметры и каким образом влияют на достижение цели. Выписать ограничения и тем самым наглядно показать, что мешает, например, получить максимальную прибыль.

Следующий важный этап состоит в построении математической модели. Как архитектор использует бумагу для построения макета здания, так и математический язык используется для записи моделей. Может оказаться, что символическая запись слишком сложна в той постановке, которая получилась на предыдущем шаге. Сложность может заключаться в том, что критерии оптимизации или ограничения выражаются нелинейными функциями или число ограничений и переменных слишком велико. В результате время работы алгоритма с такой моделью окажется неприемлемо большим. В этом случае необходимо вернуться на шаг уяснения и формулировки задачи, переопределить переменные и ограничения. Построение модели — это своего рода искусство. Не существует единственного верного способа, как это делать. Для одной и той же задачи можно построить несколько эквивалентных моделей с разным числом переменных и ограничений, так же как для теорем может быть предложено несколько различных доказательств.

Полезность модели зависит еще и от того, кто с ней работает. ЛПР должно обладать долей интуиции и понимать степень реалистичности получаемого решения. Не всегда цель моделирования состоит исключительно в том, чтобы найти *оптимальное решение задачи*, т. е. наилучшее решение для конкретной модели, но также может быть интересным исследование различных альтернативных решений, с помощью которых можно прогнозировать сложные ситуации. Не всегда оптимальное решение является подходящим для практического внедрения. Может оказаться, что поиск оптимума требует больших вычислительных ресурсов, в то время как приближенное решение не сильно отличается от оптимального или оно устраивает того, кто принимает решение. Кроме того, при моделировании часто возникают труднореализуемые аспекты. Имеется много критериев оптимизации, и понятие оптимального решения является условным. Многовариантные расчеты по сценарию «А что, если...?» позволяют проверить чувствительность решений к изменениям исходных данных, исследовать различные предположения и оценить последствия принимаемых решений.

В большинстве задач выбора имеется много критериев оценки вариантов решения. Критерии могут быть зависимыми или независимыми. Зависимыми называют те критерии, при которых оценка альтернативы по одному из них определяет оценку по другому критерию. Количество критериев влияет на сложность задачи принятия решений. При небольшом числе критериев (два-три) задача сравнения по ним может быть выполнена непосредственно. При большом числе критериев задача становится малообозримой, и тогда необходимо использовать специальные методы решения многокритериальных задач оптимизации [7, 27].

В пособии будут рассмотрены задачи, в которых поиск решения проводится по одному критерию оптимизации. Материал, содержащийся в пособии, является частью лекционных курсов и семинарских занятий по дисциплине «Теория принятия решений» [1], читаемой студентам 3 и 5-го курсов на факультете информационных технологий Новосибирского государственного университета. Программа этой дисциплины составлена в соответствии с требованиями ФГОС ВПО к структуре и результатам освоения основных образовательных программ магистратуры по циклу «Общие математические и естественно-научные дисциплины» по направлению подготовки «Информатика и вычислительная техника», а также задачами, стоящими перед Новосибирским государственным университетом по реализации Программы развития НГУ.

2. Моделирование с помощью булевых переменных

В общем виде оптимизационная модель состоит из целевой функции и ограничений. Целевая функция устанавливает зависимость критерия оптимизации от управляемых и неуправляемых параметров x и k соответственно в виде функции $f(x,k)$. Функция может быть задана аналитически или с помощью алгоритма. Для $f(x,k)$ указывается, по какому критерию будет проводиться оптимизация, минимизация или максимизация:

$$\min f(x,k) \text{ или } \max f(x,k). \tag{2.1}$$

Все ограничения модели записываются с помощью некоторых функций, также зависящих от управляемых и неуправляемых параметров x и k в виде неравенств:

$$g_i(x,k) \leq 0, \quad i = 1, \ldots, n. \tag{2.2}$$

Как правило, переменные в задачах принимают значения из заданного стандартного множества X. Например, из множества положительных действительных чисел. Поэтому в модели возникают требования на значения управляемых параметров:

$$x \in X. \tag{2.3}$$

Итак, получаем общий вид математической модели (2.1)—(2.3).

Если все взаимосвязи между управляемыми и неуправляемыми параметрами выражаются с помощью линейных функций, то специалист имеет дело с линейными моделями. Их основное преимущество — это возможность применения линейного программирования. Современные программные средства для решения задач комбинаторной оптимизации так или иначе используют метод ветвей и границ, методы отсечений и др. Очень часто эти методы проще реализуются для моделей с линейными зависимостями между параметрами и переменными.

Переменная называется *булевой* в честь английского математика Дж. Буля, если она принимает одно из двух значений 0 или 1. Иногда некоторые условия легко и естественным образом могут выражаться с помощью нелинейных функций. В этом случае для линеаризации приходится вводить дополнительные переменные, значения которых полностью определяются исходными переменными принятия решений; выписывать с помощью линейных функций ограничения, связывающие дополнительные переменные с исходными. Способы линеаризации будут рассмотрены далее в разд. 2.6.

В разд. 2.2 будут представлены правила построения линейных неравенств и равенств с булевыми переменными для выражения логических взаимосвязей между событиями. Слово «правило» не должно вызывать противоречия со словом «искусство» применительно к моделированию, поскольку некоторые общие закономерности при моделировании все же существуют.

Прежде чем переходить к моделированию с помощью булевых переменных, заметим, что любую переменную, которая принимает целое неотрицательное значение, ограниченное сверху, можно представить как сумму булевых переменных с соответствующими коэффициентами. Например, пусть переменная x принимает целые значения из отрезка $[0, U]$. Тогда x можно представить, используя двоичное разложение следующим образом:

$$x = \sum_{j=0}^{\lfloor log_2 U \rfloor} 2^j x_j,$$

где $x_j \in \{0, 1\}$ для всех $j = 0, \ldots, \lfloor log_2 U \rfloor$.

Кроме построения модели, может возникнуть обратная задача, в которой по математической модели необходимо понять содержательный смысл ограничений и взаимосвязей между переменными. Например, пусть имеются булевы переменные x_1, x_2, x_3, непрерывная переменная y, принимающая значения из отрезка $[0, 1]$, и неравенство

$$x_1 - x_2 + x_3 + y \leq 2.$$

Возникает вопрос, какие взаимосвязи между переменными реализованы в этом неравенстве? После недолгих размышлений нетрудно заметить, что это неравенство моделирует следующую зависимость: если $x_1 = 1$, $x_2 = 0$, $x_3 = 1$, то $y \leq 0$, учитывая, что $y \in [0, 1]$, следует, что $y = 0$. Моделирует ли это неравенство еще какие-либо зависимости между этими переменными? Ответ «нет», а почему это так, будет объясняться ниже.

2.1. Примеры математических моделей

Прежде чем переходить к специальным приемам и разным способам моделирования, рассмотрим несколько простых с точки зрения построения математических моделей примеров хорошо известных классических задач.

Задача о 0–1 рюкзаке

Имеется n предметов и рюкзак грузоподъемностью V. Известна полезность c_j и вес v_j каждого предмета, $j = 1, \ldots, n$. Требуется определить, какие предметы положить в рюкзак, чтобы суммарная полезность выбранных предметов была наибольшей, а общий вес не превышал грузоподъемность рюкзака [29].

Для моделирования событий «предмет положили в рюкзак» или «не положили» введем n булевых переменных:

$$x_j = \begin{cases} 1, & \text{если предмет } j \text{ положили в рюкзак,} \\ 0, & \text{если предмет } j \text{ не положили в рюкзак.} \end{cases}$$

Целевая функция задачи — максимальная суммарная полезность выбранных предметов:

$$\max \sum_{j=1}^{n} c_j x_j. \tag{2.4}$$

Тому, чтобы положить все предметы в рюкзак и таким образом достигнуть максимальной полезности, препятствует ограничение на общий вес выбранных предметов. Он не должен превышать грузоподъемность рюкзака:

$$\sum_{j=1}^{n} v_j x_j \leq V. \tag{2.5}$$

Ограничения на значения переменных:

$$x_j \in \{0, 1\}, j = 1, \ldots, n. \tag{2.6}$$

Итак, получили математическую модель (2.4)–(2.6) для задачи о 0–1 рюкзаке с n булевыми переменными и одним ограничением.

Задача о назначениях

Имеется n рабочих и m работ, $n \geq m$. Известны (c_{ij}) затраты при выполнении работы j рабочим i. Требуется составить план назначений рабочих на работы так, чтобы все работы были выполнены с минимальными суммарными затратами.

Введем nm булевых переменных:

$$x_{ij} = \begin{cases} 1, & \text{если рабочий } i \text{ выполняет работу } j, \\ 0- & \text{в противном случае.} \end{cases}$$

Целевая функция — минимальные суммарные затраты:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij}. \tag{2.7}$$

При условиях, что каждая работа должна быть выполнена

$$\sum_{i=1}^{n} x_{ij} = 1, j = 1, \ldots, m, \tag{2.8}$$

каждый рабочий может выполнять не более одной работы

$$\sum_{j=1}^{m} x_{ij} \leq 1, i = 1, \ldots, n \tag{2.9}$$

и при ограничениях на значения переменных

$$x_{ij} \in \{0, 1\}, i = 1, \ldots, n, j = 1, \ldots, m. \tag{2.10}$$

В литературе можно встретить разные интерпретации этой задачи. Она полиномиально разрешима, в [1, 13] для случая с $n = m$ описывается алгоритм, трудоемкость которого составляет $O(n^3)$.

Задача о размещении студентов в общежитии

Рассмотрим задачу, близкую к задаче о назначениях. Имеется $2n$ студентов, которых нужно расселить по n двухместным комнатам, т. е. каждому студенту назначить ровно одного соседа. Величина (c_{ij}) задает расстояние между городами, в которых жили студенты i и j до поступления в университет. Требуется разместить студентов по комнатам таким образом, чтобы перезнакомить как можно больше студентов из наиболее удаленных друг от друга городов.

Введем булевы переменные:

$$x_{ij} = \begin{cases} 1, & \text{если студенты } i \text{ и } j \text{ живут в одной комнате,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Целевая функция — максимальное расстояние между познакомившимися студентами

$$\max \sum_{i=1}^{2n-1} \sum_{j=i+1}^{2n} c_{ij} x_{ij}, \tag{2.11}$$

при ограничениях, что у каждого студента ровно один сосед

$$\sum_{k<i} x_{ki} + \sum_{j>i} x_{ij} = 1, i = 1, \ldots, 2n, \tag{2.12}$$

и ограничениях на значения переменных

$$x_{ij} \in \{0, 1\}, i = 1, \ldots, 2n, j = 1, \ldots, 2n. \tag{2.13}$$

В литературе задача (2.11)–(2.13) известна как *задача о совершенном паросочетании* [8, 13]. *Паросочетанием* в графе G с множеством вершин V и множеством ребер E называется подмножество ребер $M \subseteq E$, такое, что для всех вершин v из V в M содержится не более одного ребра, инцидентного v. Вершина v из V называется *связанной паросочетанием*, если в M есть ребро, инцидентное v. *Совершенное паросочетание* — это паросочетание, в котором каждая вершина является связанной.

В задаче (2.11)–(2.13) требовалось найти совершенное паросочетание максимального веса на двудольном графе (рис. 2).

Если в (2.12) знак равенства заменить на неравенство меньше либо равно, то получим задачу о поиске паросочетания максимального веса. Задача о назначениях с $n > m$, рассмотренная выше, эквивалентна задаче о поиске паросочетания минимального веса на двудольном графе (рис. 3).

2.2. Правила моделирования логических импликаций

Рассмотрим некоторые правила, следуя которым можно моделировать логические импликации с помощью линейных ограничений и булевых переменных [26].

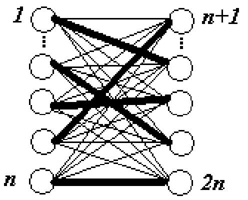

Рис. 2. Совершенное паросочетание для задачи о размещении студентов

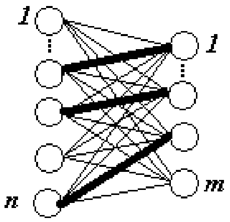

Рис. 3. Паросочетание минимального веса для задачи о назначениях

Напомним, что под *импликацией* понимается логическая связка некоторого условия и следствия из него и выражается союзами «если..., то...».

Первое правило. Пусть I — конечное множество индексов, x_i— булева переменная, $i \in I$ и y — непрерывная переменная, такая, что $0 \leq y \leq 1$. Тогда импликация

$$\text{если } x_i = 0 \text{ для всех } i \in I, \text{ то } y = 0$$

моделируется неравенством

$$y \leq \sum_{i \in I} x_i. \tag{2.14}$$

Нетрудно проверить, что неравенство (2.14) действительно реализует нужную логическую связку. Если $x_i = 0$ для всех $i \in I$, то $\sum_{i \in I} x_i = 0$ и неравенство (2.14) превращается в неравенство $y \leq 0$. Поскольку y — неотрицательная величина, то $y = 0$.

Более того, неравенство (2.14) не порождает никаких лишних ограничений, т. е. если $x_i = 1$ для некоторого i, то $1 \leq \sum_{i \in I} x_i$, и так как $y \leq 1$, то неравенство

(2.14) всегда выполнено.

Следствие 1.1. Пусть значение переменной y ограниченно сверху величиной c, $0 \le y \le c$. Тогда импликация

$$\text{если } x_i = 0 \text{ для всех } i \in I, \text{ то } y = 0$$

моделируется неравенством

$$y \le c \sum_{i \in I} x_i. \tag{2.15}$$

Пример. Задача размещения производства [9]

Задано конечное множество возможных мест производства I некоторой однородной продукции и конечное множество клиентов J. Известны затраты c_i на организацию производства в пункте i. Продукция доставляется клиентам, стоимость доставки клиенту j из пункта i равна d_{ij}. Каждый клиент может обслуживаться только из одного пункта. Необходимо определить, в каких пунктах следует разместить производство, чтобы обслужить всех клиентов с наименьшими суммарными затратами.

Введем булевы переменные:

$$x_i = \begin{cases} 1, & \text{если в пункте } i \text{ размещается производство,} \\ 0 \; - & \text{в противном случае} \end{cases}$$

и

$$y_{ij} = \begin{cases} 1, & \text{если клиент } j \text{ обслуживается из} \\ & \text{пункта производства } i, \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Первая группа ограничений должна гарантировать, что каждый клиент будет обслуживаться ровно одним открытым предприятием:

$$\sum_{i \in I} y_{ij} = 1, \text{ для каждого } j \in J. \tag{2.16}$$

Вторая группа ограничений гарантирует, что если в пункте i производство не размещено, то клиент j из него не обслуживается. В виде импликации это условие записывается так:

$$\text{если } x_i = 0, \text{ то } y_{ij} = 0, \text{ для каждого } i \in I, j \in J.$$

Следуя первому правилу, получаем:

$$y_{ij} \le x_i, \text{ для каждого } i \in I, j \in J. \tag{2.17}$$

Выпишем целевую функцию. Для этого вычислим суммарные затраты, которые нужно минимизировать. Они складываются из затрат на организацию производства и суммарных затрат на обслуживание клиентов из открытых пунктов производства:

$$\min \sum_{i \in I} c_i x_i + \sum_{i \in I} \sum_{j \in J} d_{ij} y_{ij}. \tag{2.18}$$

Получили математическую модель (2.16)–(2.18) с булевыми переменными x_i и y_{ij}.

На практике предприятия не могут производить, а клиенты не могут потреблять бесконечное количество продукции. Поэтому более реалистичная ситуация, когда производственные мощности предприятий и спрос у клиентов ограничены.

Задача размещения с ограничениями на мощности производства [9]

Пусть производственная мощность предприятия i составляет u_i единиц, а величина b_j задает спрос клиента j. Величины d_{ij} задают удельные затраты на доставку продукции от клиента j до пункта i. Изменим смысл переменных y_{ij}. Теперь они будут обозначать количество продукции, поставляемое клиенту j из пункта i. Тогда задача размещения с ограничениями на мощности производства может быть записана в следующем виде:
целевая функция

$$\min \sum_{i \in I} c_i x_i + \sum_{i \in I} \sum_{j \in J} d_{ij} y_{ij}, \tag{2.19}$$

ограничения, гарантирующие удовлетворение спроса для каждого клиента,

$$\sum_{i \in I} y_{ij} = b_j, \text{ для каждого } j \in J. \tag{2.20}$$

В модели не хватает ограничений, которые гарантировали бы, что если в пункте i открыто предприятие, то общее количество продукции, отправленное из него всем клиентам, не может быть больше, чем величина u_i, т. е.

$$\text{если } x_i = 0, \text{ то } y_{ij} = 0, \text{ для каждого } j \in J$$

и $0 \leq \sum_{j \in J} y_{ij} \leq u_i$. Модифицируя первое правило, эти условия моделируются неравенствами:

$$\sum_{j \in J} y_{ij} \leq u_i x_i, \text{ для каждого } i \in I. \tag{2.21}$$

Добавив ограничения на значения переменных:

$$x_i \in \{0, 1\}, i \in I \tag{2.22}$$

$$y_{ij} \geq 0, i \in I, j \in J, \tag{2.23}$$

получаем математическую модель (2.19)–(2.23) смешанного целочисленного линейного программирования.

Пример. Задача о покрытии

Задано конечное множество клиентов J и конечное множество пунктов размещения магазинов I. Известны кратчайшие расстояния d_{ij} между элементами i, j из множеств I и J соответственно. Величина s_j задает максимальное расстояние, которое клиент j согласен преодолеть до магазина. Учитывая это, для каждого клиента j можно определить множество магазинов N_j, которые он мог бы посещать: $N_j := \{i \in I | d_{ij} \leq s_j\}$.

Требуется определить минимальное количество магазинов, которое нужно открыть, чтобы обслужить всех клиентов. Другими словами, требуется выбрать минимальное по числу элементов подмножество множества I, покрывающее все множество J.

Введем булевы переменные:

$$x_i = \begin{cases} 1, & \text{если в } i\text{-м пункте магазин открыт,} \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Тогда целевая функция — минимизировать общее число открываемых предприятий:

$$\min \sum_{i \in I} x_i. \tag{2.24}$$

Ограничения, гарантирующие, что каждый клиент будет посещать подходящий магазин:

$$\sum_{i \in N_j} x_i \geq 1, \quad j \in J. \tag{2.25}$$

Ограничения на значения переменных:

$$x_i \in \{0, 1\}, \quad i \in I. \tag{2.26}$$

Получаем математическую модель (2.24)–(2.26) для одной из задач о покрытии. Существуют и другие постановки задач о покрытии, которые образуют целый класс этих задач [21, 30].

В некоторых задачах из этого класса покрытие каждого объекта связано с определенными затратами, а общий бюджет ограничен, поэтому покрыть множество J полностью невозможно. В этом случае решают задачу о поиске максимального количества покрытых объектов.

Пусть w_i — затраты, связанные с открытием магазина в пункте i, B — общий бюджет на открытие магазинов. Требуется обслужить как можно больше клиентов при ограниченном бюджете на открытие магазинов.

Введем новые булевы переменные:

$$y_j = \begin{cases} 1, & \text{если клиент } j \text{ обслуживается,} \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Целевая функция — максимальное число обслуживаемых клиентов:

$$\max \sum_{j \in J} y_j. \tag{2.27}$$

Переменные y_j и x_i логически связаны между собой следующим образом:

$$y_j = 1, \text{ тогда и только тогда, когда } x_i = 1 \text{ для некоторого } i \in N_j.$$

Это условие содержит две импликации. Смоделируем отдельно каждую из них. Первая импликация:

$$\text{если } x_i = 1 \text{ для некоторого } i \in N_j, \text{ то } y_j = 1$$

или

$$\text{если } x_i = 0 \text{ для всех } i \in N_j, \text{ то } y_j = 0.$$

Следуя первому правилу, получаем:

$$y_j \leq \sum_{i \in N_j} x_i \quad j \in J. \tag{2.28}$$

Вторая импликация:

$$\text{если } y_j = 1, \text{ то } x_i = 1 \text{ для некоторого } i \in N_j$$

или

$$\text{если } y_j = 0, \text{ то } x_i = 0 \text{ для всех } i \in N_j$$

Применяя первое правило, получаем формулировку этой импликации в виде следующего неравенства:

$$x_i \leq y_j, \text{ для всех } i \in N_j, \quad j \in J. \tag{2.29}$$

Суммарные затраты на открытие магазинов ограничены:

$$\sum_{i \in I} w_i x_i \leq B. \tag{2.30}$$

Ограничения на значения переменных:

$$x_i, y_j \in \{0, 1\}, \quad i \in I, j \in J. \tag{2.31}$$

Второе правило. Пусть x_i — булева переменная, $i \in I$, где I — конечное множество индексов; I_0 и I_1 — непересекающиеся подмножества множества I и y — целочисленная, или непрерывная, переменная, удовлетворяющая неравенству $0 \leq y \leq 1$. Тогда логическая импликация

$$\text{если } x_i = 0 \text{ для всех } i \in I_0 \text{ и } x_i = 1 \text{ для всех } i \in I_1, \text{ то } y = 0$$

моделируется неравенством

$$y \leq \sum_{i \in I_0} x_i + \sum_{i \in I_1} (1 - x_i). \tag{2.32}$$

Логическая импликация

$$\text{если } x_i = 0 \text{ для всех } i \in I_0 \text{ и } x_i = 1 \text{ для всех } i \in I_1, \text{ то } y = 1$$

моделируется неравенством

$$(1-y) \leq \sum_{i\in I_0} x_i + \sum_{i\in I_1}(1-x_i). \tag{2.33}$$

Эквивалентность импликаций и неравенств нетрудно проверить самостоятельно, пользуясь первым правилом.

После приведения подобных неравенства (2.32) и (2.33) можно переписать в виде

$$\sum_{i\in I_1} x_i - \sum_{i\in I_0} x_i + y \leq |I_1| \tag{2.34}$$

и

$$\sum_{i\in I_1} x_i - \sum_{i\in I_0} x_i - y \leq |I_1| - 1. \tag{2.35}$$

Этот вид более компактный, но по нему сложнее восстановить логические взаимосвязи между переменными.

Итак, *если импликация содержит $x_i = 1$ для некоторого i, то в неравенство вместо переменной x_i будет входить* $(1-x_i)$.

Пример

Смоделируем импликацию:

$$\text{если } x_1 = 1 \text{ и } x_2 = 0 \text{ и } x_3 = 1, \text{ то } y = 0.$$

Следуя второму правилу, получаем:

$$y \leq (1-x_1) + x_2 + (1-x_3) \tag{2.36}$$

или

$$x_1 - x_2 + x_3 + y \leq 2.$$

Это неравенство рассматривалось в разд. 2, и теперь должно быть понятно, какие взаимосвязи оно реализует.

Пример. Выбор ближайшего открытого предприятия

Как правило, в задачах размещения предприятий среди всех открытых предприятий клиенты выбирают для себя ближайшее. Иногда такие назначения происходят автоматически, если целевая функция направлена на минимизацию суммарного расстояния пройденного клиентами. Однако если целевая функция другая или в задаче присутствует ограничение, например, на бюджет проекта, то в оптимальном решении клиенты не обязательно будут назначены к ближайшим предприятиям. И тогда возникает необходимость моделировать в виде ограничения условие: *если в пункте s размещено предприятие и другие открытые предприятия дальше, чем s по отношению к клиенту i, то клиент i должен обслуживаться из предприятия s.*

Для моделирования введем следующие булевы переменные:

$$x_s = \begin{cases} 1, & \text{если предприятие размещено в пункте } s, \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$y_{is} = \begin{cases} 1, & \text{если клиент } i \text{ назначен к предприятию } s, \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Обозначим через C_{is} множество всех предприятий, которые находятся к клиенту i ближе, чем предприятие s. Тогда

$$\text{если } x_s = 1 \text{ и } x_t = 0 \text{ для всех } t \in C_{is}, \text{ то } y_{is} = 1.$$

Применяя второе правило, получаем:

$$1 - y_{is} \leq (1 - x_s) + \sum_{t \in C_{is}} x_t$$

или

$$y_{is} \geq x_s - \sum_{t \in C_{is}} x_t.$$

2.3. Моделирование свойств логических отношений

Моделирование отношения транзитивности. Часто в задачах между объектами заданного множества I определено некоторое отношение R. Например, $i < j$, в данном случае под R понимается отношение сравнения «строго меньше». Как правило, R должно обладать свойством транзитивности, т. е. если iRj и jRk, то iRk для любых $i, j, k \in I$.

Рассмотрим граф с множеством вершин V и множеством ребер E. Требуется смоделировать транзитивность отношения связности вершин. Пусть булева переменная:

$$x_{ij} = \begin{cases} 1, & \text{если между вершинами } i \text{ и } j \text{ есть путь,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Тогда для любой тройки вершин $(i, j, k) \in V \times V \times V$

$$\text{если } x_{ij} = 1 \text{ и } x_{jk} = 1, \text{ то } x_{ik} = 1.$$

Следуя второму правилу, получаем линейное неравенство:

$$(1 - x_{ik}) \leq (1 - x_{ij}) + (1 - x_{jk})$$

или

$$x_{ij} + x_{jk} - x_{ik} \leq 1. \tag{2.37}$$

Моделирование отношения порядка. В задачах теории расписаний между работами, выполняемыми на одной машине, должны соблюдаться отношения предшествования. Очередная работа не может начаться, пока не закончится предыдущая работа. Отношения порядка, т. е. транзитивности и антисимметричности на множестве работ, в таких задачах можно моделировать по-разному. Рассмотрим вариант со следующими переменными.

Введем булевы переменные:

$$x_{ij} = \begin{cases} 1, & \text{если работа } i \text{ выполняется до работы } j, i \neq j \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Запишем условие антисимметричности. Для любой пары работ i и j всегда одна работа должна выполняться раньше другой, т. е. только одна из переменных x_{ij} или x_{ji} может быть равна единице:

$$x_{ij} + x_{ji} = 1,$$

и $x_{ii} = 0$. Следовательно, можно ограничиться рассмотрением только переменных x_{ij} с $i < j$ и $x_{ii} = 0$.

Для каждой тройки работ i, j и k должно выполняться отношение транзитивности:

$$\begin{aligned}
&\text{при } i < j < k \quad (1 - x_{ik}) \leq (1 - x_{ij}) + (1 - x_{jk}),\\
&\text{при } i < k < j \quad (1 - x_{ij}) \leq (1 - x_{ik}) + (1 - x_{kj}),\\
&\text{при } k < i < j \quad (1 - x_{kj}) \leq (1 - x_{ki}) + (1 - x_{ij}),\\
&\text{при } k < j < i \quad (1 - x_{ki}) \leq (1 - x_{kj}) + (1 - x_{ji}),\\
&\text{при } j < i < k \quad (1 - x_{jk}) \leq (1 - x_{ji}) + (1 - x_{ik}),\\
&\text{при } j < k < i \quad (1 - x_{ji}) \leq (1 - x_{jk}) + (1 - x_{ki}).
\end{aligned}$$

С учетом антисимметричности, при переходе к переменным x_{ij} с $i < j$ из шести групп неравенств останутся только следующие неравенства:

$$x_{ij} + x_{jk} - x_{ik} \leq 1 \text{ для любых } (i, j, k), i < j < k, \tag{2.38}$$

$$(1 - x_{ij}) - x_{jk} + x_{ik} \leq 1 \text{ для любых } (i, j, k), i < j < k. \tag{2.39}$$

Неравенство (2.38) гарантирует транзитивность в очередности выполнения работ i, j и k, если работа i предшествует работе j $(x_{ij} = 1)$. Если это не так, то неравенство (2.38) верно при любых значениях x_{jk} и x_{ik}, а с помощью неравенства (2.39) сохраняется транзитивность при $x_{ij} = 0$.

Моделирование отношений эквивалентности. В отличие от предыдущего примера, в некоторых задачах требуется, чтобы между объектами сохранялось отношение эквивалентности, т. е. соблюдалась и транзитивность, и симметричность, например, в задачах кластеризации, в которых некоторое конечное множество объектов I нужно разбить на подмножества. Пусть булевы переменные x_{ij} показывают, принадлежат ли объекты i и j одному подмножеству:

$$x_{ij} = \begin{cases} 1, & \text{если объекты } i \text{ и } j \text{ лежат в одном подмножестве,} \\ 0 - & \text{в противном случае.} \end{cases}$$

В силу симметричности принадлежности двух объектов одному подмножеству число переменных можно сократить и рассматривать только x_{ij} с $i < j$, добавив условия, что $x_{ji} = x_{ij}$, и условие рефлексивности $x_{ii} = 1$. Кроме того, для любой тройки должно выполняться отношение транзитивности. Но, в отличие от предыдущего примера из теории расписаний, в задачах разбиения между объектами, попавшими в одно подмножество, должна соблюдаться симметричность. Поэтому кроме одного неравенства типа (2.37) нужно сформулировать еще два неравенства. Итак, первое неравенство гарантирует, что если i и j в одном подмножестве, j и k в одном подмножестве, то i и k также в одном подмножестве, т. е.

$$x_{ij} + x_{jk} - x_{ik} \leq 1 \text{ для любых } (i, j, k), i < j < k. \tag{2.40}$$

Второе неравенство гарантирует, что если i и j в одном подмножестве, i и k в одном подмножестве, то j и k также в одном подмножестве, т. е.

$$x_{ij} + x_{ik} - x_{jk} \leq 1 \text{ для любых } (i, j, k), i < j < k. \tag{2.41}$$

Третье неравенство гарантирует, что если i и k в одном подмножестве, j и k в одном подмножестве, то i и j также в одном подмножестве, т. е.

$$x_{ik} + x_{jk} - x_{ij} \leq 1 \text{ для любых } (i, j, k), i < j < k. \tag{2.42}$$

2.4. Моделирование выбора минимального элемента

Часто в задачах нужно, чтобы переменная из нескольких значений принимала минимальное значение. Например, необходимо, чтобы переменная y принимала значение, равное $\min(u_1, u_2)$, где $u_1, u_2 \geq 0$. Значит, одновременно должны выполняться неравенства

$$y \leq u_1 \text{ и } y \leq u_2$$

и одно из неравенств

$$y \geq u_1 \text{ или } y \geq u_2.$$

Сложность возникает в моделировании выполнения одного из двух последних неравенств. Введем две вспомогательные булевы переменные:

$$x_1 = \begin{cases} 1, & \text{если неравенство } y \geq u_1 \text{ выполняется,} \\ 0 - & \text{в противном случае,} \end{cases} \tag{2.43}$$

$$x_2 = \begin{cases} 1, & \text{если неравенство } y \geq u_2 \text{ выполняется,} \\ 0 - & \text{в противном случае,} \end{cases} \tag{2.44}$$

через W обозначим некоторое большое положительное число. Тогда выбор минимального из двух чисел u_1 и u_2 эквивалентен совместности следующей системы ограничений:

$$y \leq u_1 \tag{2.45}$$

$$y \leq u_2 \tag{2.46}$$

$$x_1 + x_2 = 1 \tag{2.47}$$

$$y \geq u_1 - W(1 - x_1) \tag{2.48}$$

$$y \geq u_2 - W(1 - x_2) \tag{2.49}$$

Ограничение (2.47) говорит о том, что ровно одна из переменных, x_1 или x_2, будет равна 1. Предположим, что $u_1 < u_2$. Тогда случай, когда $x_1 = 0, x_2 = 1$, не возможен потому, что ограничение (2.48) будет выполнено при любом значении y, но из ограничения (2.49) получается $y \geq u_2$, следовательно, система (2.45)–(2.49) несовместна. Остается вариант, когда $x_1 = 1$, $x_2 = 0$, в этом случае условие (2.49) будет выполнено при любом значении y, а согласно неравенствам (2.45) и (2.48) получаем $y = u_1$.

2.5. Моделирование взаимоисключающих событий

Рассмотрим ситуацию, в которой возникает необходимость выполнения некоторой подгруппы ограничений. Предположим, что допустимая область формируется из нескольких групп неравенств. Например, пусть допустимая область соответствует заштрихованной части, как показано на рис. 4, и образована объединением двух многоугольников P_1 и P_2.

Каждый многоугольник задается группой неравенств.

P_1 :

$$2y_1 + y_2 \geq 4,$$

$$y_1 - y_2 \geq -4,$$

$$-4y_1 + 3y_2 \geq -8,$$

$$2y_1 + 3y_2 \leq 22,$$

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

P_2 :

$$3y_1 + 4y_2 \geq 22,$$

$$-3y_1 + 4y_2 \leq 10,$$

$$3y_1 + 3y_2 \leq 39,$$

$$y_1 - y_2 \leq 5,$$

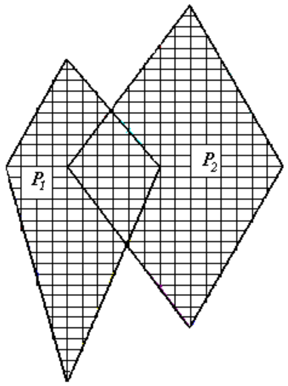

Рис. 4. Допустимая область

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

Решение является допустимым, если оно удовлетворяет хотя бы одной группе неравенств P_1 или P_2. Другими словами, из двух групп ограничений нужно, чтобы выполнялась, по крайней мере, одна группа. Рассмотрим два способа моделирования этой ситуации.

Первый способ

Преобразуем все ограничения, которые этого требуют, кроме ограничений на значение переменных, в ограничения со знаком меньше либо равно в неравенствах:

P_1 :

$$-2y_1 - y_2 \leq -4,$$

$$-y_1 + y_2 \leq 4,$$

$$4y_1 - 3y_2 \leq 8,$$

$$2y_1 + 3y_2 \leq 22,$$

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

P_2 :

$$-3y_1 - 4y_2 \leq -22,$$

$$-3y_1 + 4y_2 \leq 10,$$

$$3y_1 + 3y_2 \leq 39,$$

$$y_1 - y_2 \le 5,$$
$$y_1 \ge 0, y_2 \ge 0.$$

Введем булевы переменные:

$$x_1 = \begin{cases} 1, & \text{если выполнена группа неравенств для } P_1, \\ 0 \ - & \text{в противном случае,} \end{cases} \tag{2.50}$$

$$x_2 = \begin{cases} 1, & \text{если выполнена группа неравенств для } P_2, \\ 0 \ - & \text{в противном случае.} \end{cases} \tag{2.51}$$

Введем вспомогательный вектор (w_1, w_2, w_3, w_4) с числом компонент, равным максимальному числу неравенств в P_1 и P_2, а значения компонент — большие положительные числа, такие, что при добавлении компоненты к правой части соответствующего неравенства это неравенство выполняется при любых значениях y_1 и y_2. Выпишем следующую систему ограничений I:

$$-2y_1 - y_2 \le -4 + w_1(1 - x_1),$$
$$-y_1 + y_2 \le 4 + w_2(1 - x_1),$$
$$4y_1 - 3y_2 \le 8 + w_3(1 - x_1),$$
$$2y_1 + 3y_2 \le 22 + w_4(1 - x_1),$$

$$-3y_1 - 4y_2 \le -22 + w_1(1 - x_2),$$
$$-3y_1 + 4y_2 \le 10 + w_2(1 - x_2),$$
$$3y_1 + 3y_2 \le 39 + w_3(1 - x_2),$$
$$y_1 - y_2 \le 5 + w_4(1 - x_2),$$

$$x_1 + x_2 \ge 1, \tag{2.52}$$
$$x_1, x_2 \in \{0, 1\},$$
$$y_1 \ge 0, y_2 \ge 0.$$

Условие (2.52) означает, что, по крайней мере, одна группа ограничений для P_1 или для P_2 выполняется. Если $x_1 = 0$, то в силу (2.52) получается $x_2 = 1$, и группа ограничений для P_1 становится избыточной из-за наличия компонент вспомогательного вектора, а выполняться должна другая группа ограничений для P_2. Аналогично если $x_2 = 0$, то $x_1 = 1$, и решение должно удовлетворять условиям для P_2. Если $x_1 = 1$, $x_2 = 1$, то обе группы ограничений должны быть выполнены.

Второй способ

Увеличим число переменных следующим образом. Будем разделять переменные y_1 и y_2 на два типа. В системе неравенств, определяющей область P_1, будем использовать переменные y_1^1, y_2^1. В системе неравенств, задающих область P_2, будем использовать переменные y_1^2, y_2^2. Причем $y_1^1 + y_1^2 = y_1$, $y_2^1 + y_2^2 = y_2$. Кроме того, введем булевы переменные x_1, x_2, смысл которых, как и в первом способе. Запишем систему ограничений II :

$$
\begin{gathered}
2y_1^1 + y_2^1 \geq 4x_1, \\
y_1^1 - y_2^1 \geq -4x_1, \\
-4y_1^1 + 3y_2^1 \geq -8x_1, \\
2y_1^1 + 3y_2^1 \leq 22x_1, \\
\\
3y_1^2 + 4y_2^2 \geq 22x_2, \\
-3y_1^2 + 4y_2^2 \leq 10x_2, \\
3y_1^2 + 3y_2^2 \leq 39x_2, \\
y_1^2 - y_2^2 \leq 5x_2, \\
\\
y_1^1 + y_1^2 = y_1, \\
y_2^1 + y_2^2 = y_2, \\
x_1 + x_2 = 1, \qquad\qquad (2.53)\\
y_1^1 \geq 0, y_2^1 \geq 0, y_1^2 \geq 0, y_2^2 \geq 0, y_1 \geq 0, y_2 \geq 0, \\
x_1, x_2 \in \{0, 1\}.
\end{gathered}
$$

Утверждается, что $P_1 \cup P_2 \neq \emptyset$ тогда и только тогда, когда система II разрешима. Проверим это.

Пусть y_1, $y_2 \in (P_1 \cup P_2)$, можно считать, что y_1, $y_2 \in P_1$. Тогда полагаем $x_1 = 1$, $x_2 = 0$, $y_1 = y_1^1$, $y_2 = y_2^1$, $y_1^2 = y_2^2 = 0$ — решение системы.

Пусть, без ограничений общности, $x_1 = 1$, $x_2 = 0$, $y_1 = y_1^1$, $y_2 = y_2^1$, $y_1^2 = y_2^2 = 0$ — решение системы II. Тогда $y_1, y_2 \in P_1$, следовательно, $P_1 \cup P_2 \neq \emptyset$.

К вопросу о том, какой из двух способов лучше, вернемся позже, а пока подчеркнем, что этот пример показывает, как с помощью линейных ограничений и неравенств моделировать взаимоисключающие условия. Такие условия часто возникают в задачах составления расписаний. Например, если нужно задать порядок выполнения работ, в котором выполнение некоторой работы нельзя начать, пока не закончится другая работа.

Задача составления расписания

Имеется n работ и m машин. Для каждой работы задан порядок выполнения на машинах, т. е. работа j сначала выполняется на машине с номером

$j(1)$, затем на машине с номером $j(2)$ и т. д. В каждый момент времени машина может выполнять не более одной работы, каждая работа выполняется не более, чем на одной машине. Работы не прерываются. Известна длительность p_{ij} выполнения работы j на машине i. Требуется выполнить все работы и минимизировать сумму времен завершения всех работ.

Введем целочисленные переменные t_{ij}, означающие время начала выполнения работы j на машине i. Введем mnn булевых переменных, чтобы задать условия непересечения работ при выполнении на одной машине:

$$x_{ijk} = \begin{cases} 1, & \text{если работа } j \text{ предшествует работе } k \\ & \text{на машине } i, j < k, \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Возникают взаимоисключающие друг друга условия. Если работа j предшествует работе k на машине i, то время начала работы k должно наступить не раньше времени завершения работы j:

$$t_{ik} \geq t_{ij} + p_{ij}, \text{ если } x_{ijk} = 1$$

и

$$t_{ij} \geq t_{ik} + p_{ik}, \text{ если } x_{ijk} = 0.$$

Используя рассуждения изложенные в этом разделе, эти условия можно реализовать с помощью следующих неравенств:

$$t_{ik} \geq t_{ij} + p_{ij} - W(1 - x_{ijk}),$$

$$t_{ij} \geq t_{ik} + p_{ik} - W x_{ijk},$$

где W — большое положительное число.

Каждая работа состоит из операций, выполняемых на разных машинах, и $(r+1)$-я операция работы j не может начаться, пока не будет завершена предыдущая r-я операция, значит,

$$t_{j(r+1)j} \geq t_{j(r)j} + p_{j(r)j}.$$

Итак, математическая модель выглядит следующим образом. Целевая функция — минимальная сумма времен завершения всех работ:

$$\min \sum_{j=1}^{n} (t_{j(m)j} + p_{j(m)j}) \tag{2.54}$$

при ограничениях:

$$t_{j(r+1)j} \geq t_{j(r)j} + p_{j(r)j}, r = 1, \ldots, (m-1), j = 1, \ldots, n,$$

$$t_{ik} \geq t_{ij} + p_{ij} - W(1 - x_{ijk}), i = 1, \ldots, m; j, k = 1, \ldots, n,$$

$$t_{ij} \geq t_{ik} + p_{ik} - W x_{ijk}, i = 1, \ldots, m; j, k = 1, \ldots, n,$$

$$t_{ij} \geq 0, i = 1, \ldots, m; j = 1, \ldots, n,$$

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

Заметим, что в целевой функции (2.54) величина $\sum_{j=1}^{n} p_{j(m)j}$ является постоянной и не зависит от порядка выполнения работ. Следовательно, оптимальное решение не изменится, если целевую функцию (2.54) заменить на $\min \sum_{j=1}^{n} t_{j(m)j}$.

2.6. Линеаризация в математических моделях

2.6.1. Линеаризация произведения переменных

Если в задаче встречается квадратичное выражение вида $x_i x_j$, где $x_i, x_j \in \{0, 1\}$, то для него можно предложить эквивалентную линейную переформулировку. Для этого введем новую булеву переменную y_{ij}, такую, что $y_{ij} = x_i x_j$, т. е.

$$y_{ij} = 1 \text{ тогда и только тогда, когда } x_i = 1 \text{ и } x_j = 1,$$

другими словами,

$$\text{если } y_{ij} = 1, \text{ то } x_i = 1,$$

$$\text{если } y_{ij} = 1, \text{ то } x_j = 1$$

и

$$\text{если } x_i = 1 \text{ и } x_j = 1, \text{ то } y_{ij} = 1.$$

Применяя первое и второе правила к трем импликациям, получаем три следующих неравенства:

$$1 - x_i \leq 1 - y_{ij},$$

$$1 - x_j \leq 1 - y_{ij},$$

$$1 - y_{ij} \leq 1 - x_i + 1 - x_j.$$

Или в более упрощенной форме:

$$y_{ij} \leq x_i, \tag{2.55}$$

$$y_{ij} \leq x_j, \tag{2.56}$$

$$x_i + x_j - y_{ij} \leq 1. \tag{2.57}$$

$$y_{ij}, x_i, x_j \in \{0, 1\} \tag{2.58}$$

Интересное и полезное наблюдение с точки зрения линейного программирования заключается в том, что даже если отказаться от булевости переменных y_{ij} и перейти к непрерывным значениям из отрезка $[0, 1]$, то неравенства (2.55)–(2.58) будут выполняться, только когда переменные y_{ij} равны 0 или 1.

Пример. Задача о клике

Задан неориентированный граф G с множеством вершин V и множеством ребер E. Задача заключается в том, чтобы найти максимальный (по количеству вершин) полный подграф, т. е. клику. Напомним, что простой граф без петель и кратных ребер называется *полным*, если любая пара вершин соединена ребром.

Введем следующие булевы переменные:

$$x_v = \begin{cases} 1, & \text{если вершина } v \text{ входит в подграф,} \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Тогда целевая функция найти клику максимальной мощности:

$$\max \sum_{v \in V} x_v. \tag{2.59}$$

Подграф является кликой тогда и только тогда, когда он не содержит пару вершин, между которыми нет ребра в исходном графе, т. е.

$$\text{если } x_w = 1, \text{ то } x_v = 0, \text{ для } (v, w) \notin E.$$

С учетом описанных выше правил, эта импликация моделируется неравенством:

$$x_v \leq 1 - x_w \text{ для } (v, w) \notin E. \tag{2.60}$$

Добавив ограничения на значения переменных:

$$x_v \in \{0, 1\}, \quad v \in V, \tag{2.61}$$

получаем постановку задачи о клике в виде математической модели (2.59)–(2.61) целочисленного линейного программирования.

Задача о клике часто возникает в приложениях, связанных с телекоммуникационными, транспортными и другими сетями. В них могут быть заданы веса ребер или вершин и ищется максимальная по весу клика. Кроме того, в таких задачах, как правило, присутствует много других ограничений, среди которых — требования полноты подграфа. Целевая функциях в таких задачах может зависеть от суммарного веса ребер, входящих в клику. И тогда кроме вершин, образующих максимальную клику, нужно знать еще и ребра, входящие в полный подграф. В этой ситуации переменных, соответствующих только вершинам, недостаточно, и вводятся дополнительные булевые переменные:

$$y_{vw} = \begin{cases} 1, & \text{если ребро между вершинами } v \text{ и } w \\ & \text{входит в подграф,} \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Ребро (v, w) из множества E входит в подграф тогда и только тогда, когда обе вершины v и w входят в подграф. Следовательно, $y_{vw} = x_v x_w$. Произведение можно линеаризовать, переписав в виде линейной системы неравенств. Из импликации

$$y_{vw} = 1 \text{ тогда и только тогда, когда } x_v = 1 \text{ и } x_w = 1 \text{ для } (v, w) \in E,$$

пользуясь правилами, получаем:

$$1 - y_{vw} \le (1 - x_v) + (1 - x_w), (v, w) \in E, \tag{2.62}$$

$$y_{vw} \le x_v, (v, w) \in E, \tag{2.63}$$

$$y_{vw} \le x_w, (v, w) \in E. \tag{2.64}$$

Итак, неравенства (2.60)–(2.64) гарантируют, что выбранный подграф будет кликой.

Заметим, что можно предложить другой вариант моделирования полноты подграфа. Рассмотрим произвольные четыре вершины. Если одна пара вершин в графе соединена ребром и другая пара вершин соединена ребром, а между какими-либо вершинами из этих пар ребра в графе нет, то в подграф должна входить только одна пара из этих вершин. Формально для любых s, t, u, $v \in V$, таких, что $(s,t) \notin E$, $(s,u) \in E$ и $(t,v) \in E$,

$$\text{если } y_{su} = 1, \text{ то } y_{tv} = 0,$$

в аналитической форме

$$y_{tv} \le 1 - y_{su}. \tag{2.65}$$

Эту формулировку можно получить из неравенств (2.60)–(2.64). Просуммировав пару неравенств типа (2.63) для $(s,u) \in E$ и $(t,v) \in E$, получим $y_{su} + y_{tv} \le x_s + x_t$. Поскольку $(s,t) \notin E$, то из (2.60) получаем $x_s + x_t \le 1$, следовательно, неравенство (2.65) выполняется.

Последний вариант модели содержит меньше переменных, их столько, сколько ребер в исходном графе, но существенно больше ограничений по сравнению с (2.60)–(2.64).

2.6.2. Линеаризация заменой переменных

Пример. Задача размещения с распределенными закупками

Предпринимателю известно конечное множество I возможных мест для открытия p торговых центров и конечное множество потребителей J. Под потребителями можно понимать не индивидуальное лицо, а «типичного» представителя, который характеризует поведение некоторой группы людей. Потребители выбирают открытый торговый центр и расходуют деньги пропорционально своим предпочтениям. Предпочтение торгового центра i потребителем j измеряется величиной u_{ij}, $u_{ij} \ge 1$ для всех $i \in I$, $j \in J$. Бюджет каждого потребителя ограничен величиной B_j. Предприниматель оценивает свою прибыль от каждой потраченной потребителем j денежной единицы в магазине i величиной c_{ij}. Задача предпринимателя — открыть p торговых центров так, чтобы получить максимальную прибыль.

Введем булевые переменные:

$$x_i = \begin{cases} 1, & \text{если в месте } i \text{ открывается торговый центр,} \\ 0 \; - & \text{в противном случае} \end{cases}$$

и неотрицательные вещественные переменные y_{ij}, которые означают сумму, потраченную клиентом j в торговом центре i.

Целевая функция — максимальная суммарная прибыль:

$$\max \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} \tag{2.66}$$

при ограничениях на число открываемых предприятий:

$$\sum_{i \in I} x_i = p \tag{2.67}$$

и при условиях, что каждый потребитель тратит свой бюджет во всех открытых торговых центрах пропорционально предпочтениям:

$$y_{ij} = B_j \frac{u_{ij} x_i}{\sum_{k \in I} u_{kj} x_k}, i \in I, j \in J, \tag{2.68}$$

$$x_i \in \{0, 1\}, y_{ij} \geq 0, i \in I, j \in J. \tag{2.69}$$

Недостатком этой модели является нелинейная связь между переменными x_i и y_{ij} в ограничениях (2.68).

Линеаризация модели

Сделаем замену переменных. Введем новые неотрицательные вещественные переменные z_j, $j \in J$, такие, что

$$z_j = \frac{B_j}{\sum_{k \in I} u_{kj} x_k}. \tag{2.70}$$

Тогда целевая функция и ограничение на число открываемых предприятий остаются без изменений. Условия, что клиенты могут посещать торговый центр, только если он открыт, записывается следующим образом:

$$y_{ij} \leq B_j x_i, i \in I, j \in J. \tag{2.71}$$

Потребители тратят весь свой бюджет:

$$\sum_{i \in I} y_{ij} = B_j, j \in J. \tag{2.72}$$

Следующая группа ограничений устанавливает связь между переменными y_{ij} и z_j и определяет значения переменных z_j в соответствии с равенством (2.70):

$$y_{ij} \leq u_{ij} z_j, i \in I, j \in J, \tag{2.73}$$

$$u_{ij} z_j \leq y_{ij} + B_j(1 - x_i), i \in I, j \in J, \tag{2.74}$$

ограничения на значения переменных:

$$y_{ij} \geq 0, z_j \geq 0, x_i \in \{0,1\}, i \in I, j \in J. \tag{2.75}$$

Группы ограничений (2.73), (2.74) означают, что если i-й торговый центр открыт, т. е. $x_i = 1$, то значение y_{ij} будет совпадать со значением $u_{ij}z_j$. Если же i-й торговый центр закрыт, $x_i = 0$, то благодаря наличию в ограничении (2.74) достаточно большого числа B_j неравенство остается верным.

Пример. Задача о ценообразовании

Фирма-производитель предлагает потребителям однородную продукцию в нескольких филиалах по разным ценам. Каждый потребитель выбирает наиболее выгодный филиал, учитывая транспортные затраты до него, свой бюджет и стоимость продукции в филиале. Если сумма транспортных затрат и стоимости продукции превышает бюджет клиента, то клиент не покупает продукцию. Если же бюджета хватает на покупку и несколько филиалов дают минимальные затраты для клиента, то клиент выбирает филиал с минимальными транспортными затратами. Задача фирмы — назначить такую стоимость продукции в каждом филиале, чтобы получить максимальный суммарный доход от своих филиалов.

Обозначим через I множество филиалов, через J множество потребителей; b_j — бюджет j-го потребителя; c_{ij} — транспортные затраты от i-го филиала до j-го потребителя.

Введем неотрицательные переменные p_i — стоимость продукции в i-м филиале и булевые переменные:

$$x_{ij} = \begin{cases} 1, & \text{если } j\text{-й потребитель выбрал } i\text{-й филиал,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Целевая функция задачи — максимальный суммарный доход фирмы:

$$\max \sum_{i \in I} p_i \sum_{j \in J} x_{ij}. \tag{2.76}$$

Каждый потребитель выбирает не более одного филиала:

$$\sum_{i \in I} x_{ij} \leq 1, \quad j \in J. \tag{2.77}$$

Затраты каждого потребителя не должны превышать его бюджет:

$$\sum_{i \in I} (b_j - c_{ij} - p_i) x_{ij} \geq 0, \quad j \in J. \tag{2.78}$$

Из всех филиалов каждый потребитель выбирает тот филиал, который доставляет минимальные суммарные затраты, включающие в себя транспортные расходы и расходы на покупку продукции:

$$\sum_{i \in I} (c_{ij} + p_i) x_{ij} \leq c_{kj} + p_k, \quad k \in I, \ j \in J. \tag{2.79}$$

Ограничения на принимаемые значения переменных:

$$p_i \geq 0, x_{ij} \in \{0, 1\}, \quad i \in I,\ j \in J. \tag{2.80}$$

Недостатком этой модели является нелинейность целевой функции (2.76) и ограничений (2.78), (2.79).

Линеаризация модели для задачи о ценообразовании

Обозначим через $\overline{p}_i$ максимально возможную цену в i-м филиале: $\overline{p}_i = \max_{j \in J}(b_j - c_{ij})$. Введем неотрицательные вещественные переменные z_{ij}, отвечающие за доход, который получает производитель от i-го филиала и j-го потребителя в нем: $z_{ij} = p_i x_{ij}$. Используя данные обозначения, получим следующую модель:

$$\max \sum_{i \in I} \sum_{j \in J} z_{ij} \tag{2.81}$$

при ограничениях:

$$\sum_{i \in I} (b_j - c_{ij}) x_{ij} - \sum_{i \in I} z_{ij} \geq 0, \quad j \in J, \tag{2.82}$$

$$c_{kj} + p_k - \sum_{i \in I} c_{ij} x_{ij} - \sum_{i \in I} z_{ij} \geq 0, \quad k \in I,\ j \in J, \tag{2.83}$$

$$(1 - x_{ij})\overline{p}_i - z_{ij} + p_i \geq 0, \quad i \in I,\ j \in J, \tag{2.84}$$

$$(1 - x_{ij})\overline{p}_i + z_{ij} - p_i \geq 0, \quad i \in I,\ j \in J, \tag{2.85}$$

$$z_{ij} \leq \overline{p}_i x_{ij}, \quad i \in I,\ j \in J, \tag{2.86}$$

$$\sum_{i \in I} x_{ij} \leq 1, \quad j \in J, \tag{2.87}$$

$$p_i \geq 0, x_{ij} \in \{0, 1\}, z_{ij} \geq 0, \quad i \in I,\ j \in J. \tag{2.88}$$

Ограничения (2.84)–(2.86) гарантируют, что доход фирмы z_{ij} от обслуживания j-го потребителя в i-м филиале равен p_i, если j-й потребитель выбрал i-й филиал, т. е. $x_{ij} = 1$, и равен нулю из условий (2.86) при $x_{ij} = 0$. По сути, ограничения (2.84)–(2.86) заменяют равенство $z_{ij} = p_i x_{ij}$, что и приводит к линейной модели.

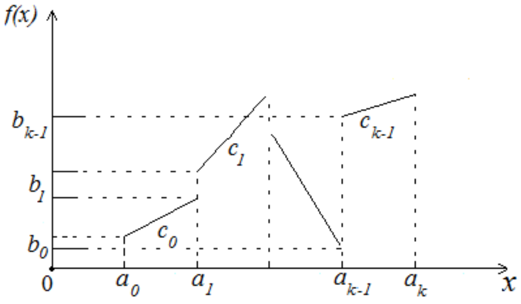

Рис. 5. Кусочно-линейная функция

2.6.3. Линеаризация кусочно-линейной функции

Пусть на отрезке $[a_0, a_k]$ задана кусочно-линейная функция (рис. 5):

$$f(x) = \begin{cases} 0, \text{ если } x = a_0, \\ b_0 + c_0(x - a_0), \text{ если } a_0 < x \le a_1, \\ b_0 + c_0(a_1 - a_0) + b_1 + c_1(x - a_1), \text{ если } a_1 < x \le a_2, \\ \vdots \\ \sum_{i=1}^{k-1}[b_{i-1} + c_{i-1}(a_i - a_{i-1})] + b_{k-1} + c_{k-1}(x - a_{k-1}), \\ \text{если } a_{k-1} < x \le a_k, \end{cases}$$

Коэффициенты b_i, c_i, $i = 0, \ldots, (k-1)$ — действительные числа. Требуется смоделировать поиск минимума функции $f(x)$, при $a_0 \le x \le a_k$.

Введем булевые переменные:

$$y_i = \begin{cases} 1, & \text{если } x > a_i \\ 0 \ - & \text{в противном случае} \end{cases}$$

и неотрицательные переменные z_i — столько от числа x лежит в интервале $[a_i, a_{i+1}]$, т. е. $z_i = x - a_i$.

MIP для поиска минимума $f(x)$ выглядит следующим образом:

$$\min \sum_{i=0}^{k-1}(b_i y_i + c_i z_i)$$

при ограничениях:

$$(a_1 - a_0)y_1 \le z_0 \le (a_1 - a_0)y_0,$$

$$(a_2 - a_1)y_2 \le z_1 \le (a_2 - a_1)y_1,$$
$$\vdots$$
$$0 \le z_{k-1} \le (a_k - a_{k-1})y_{k-1},$$
$$x = \sum_{i=0}^{k-1} z_i + a_0,$$
$$x \ge 0, y_i \in \{0,1\}, z_i \ge 0, i = 0, \ldots, (k-1).$$

Проверим, что если $(\overline{y}, \overline{z}, \overline{x})$ — допустимое решение, удовлетворяющее ограничениям в этой модели, то $f(\overline{x}) = \sum_{i=0}^{k-1}(b_i\overline{y}_i + c_i\overline{z}_i)$.

Рассмотрим $\overline{y}$. Предположим, что $\overline{y}_{r-1} = 1$, $\overline{y}_r = 0$, $1 \le r \le (k-1)$. Т.к. $\overline{y}_r = 0$, то

$$\begin{cases} \overline{y}_i = 0 & \text{при } i \ge r, \\ \overline{z}_i = 0 & \text{при } i \ge r, \end{cases}$$

т.к. $\overline{y}_{r-1} = 1$, то

$$\begin{cases} \overline{y}_i = 1 & \text{при } i \le (r-1), \\ \overline{z}_{i-1} = a_i - a_{i-1} & \text{при } i \le (r-1), \\ 0 \le \overline{z}_{i-1} \le a_i - a_{i-1} & \text{при } i = r. \end{cases}$$

Следовательно, $\sum_{i=0}^{k-1}(b_i\overline{y}_i + c_i\overline{z}_i) = \sum_{i=0}^{r-1} b_i + \sum_{i=1}^{r-1} c_{i-1}(a_i - a_{i-1}) + c_{r-1}\overline{z}_{r-1} = \sum_{i=0}^{r-1} b_i + \sum_{i=1}^{r-1} c_{i-1}(a_i - a_{i-1}) + c_{r-1}(\overline{x} - a_{r-1})$. Последнее равенство следует из того, что по построению $\overline{x} = a_{r-1} + \overline{z}_{r-1}$ при $0 \le \overline{z}_{r-1} \le (a_r - a_{r-1})$, следовательно $f(\overline{x}) = \sum_{i=0}^{k-1}(b_i\overline{y}_i + c_i\overline{z}_i)$.

Заметим, что необходимость в моделировании кусочно-линейной функции линейными ограничениями может возникнуть, когда целевая функция задачи более сложная и представляет собой линейную функцию, включающую в себя кусочно-линейную.

2.7. Симметрия в математических моделях

Один из недостатков, которым могут обладать математические модели, является симметрия в допустимых решениях. Этот недостаток приводит к тому, что точные методы, типа ветвей и границ, будут тратить время на просмотр и проверку эквивалентных решений. Рассмотрим один из вариантов симметрии и как от нее избавиться на примере задачи кластеризации [26].

Пусть требуется разбить конечное множество объектов I на p групп. Каждый объект может попасть только в одну группу. В разд. 2.3 уже рассматривались задачи кластеризации, но переменные, которые использовались там, в данном случае будут неудобными, поскольку сейчас необходимо отслеживать количество групп. Введем другие булевы переменные:

$$x_{ik} = \begin{cases} 1, & \text{если объект } i \text{ попадает в группу } k, \\ 0 - & \text{в противном случае,} \end{cases}$$

где $i \in I$, $k = 1, \ldots, p$. Каждый объект должен попасть только в одну группу, значит,

$$\sum_{k=1}^{p} x_{ik} = 1. \qquad (2.89)$$

Сложность возникает в том, что группы можно пронумеровать по-разному, и в результате будут получаться формально разные решения, но по содержащимся в группах объектам это будет одно и то же решение. Например, для $I = \{a, b, c, d\}$, $p = 3$ и разбиения $\{a\}$, $\{b, d\}$, $\{c\}$ следующие решения эквивалентны:

Таблица 4

Эквивалентные решения

Группа 1	Группа 2	Группа 3
$\{b, d\}$	$\{a\}$	$\{c\}$
$\{b, d\}$	$\{c\}$	$\{a\}$
$\{a\}$	$\{b, d\}$	$\{c\}$
$\{a\}$	$\{c\}$	$\{b, d\}$
$\{c\}$	$\{b, d\}$	$\{a\}$
$\{c\}$	$\{a\}$	$\{b, d\}$

Таким образом, каждому разбиению множества объектов соответствует $(p!)$ эквивалентных решений, отличающихся друг от друга нумерацией групп. Возникает необходимость избавиться от перестановочной симметрии и лишних, по сути, решений, оставив только одно из них. Одним из способов может быть установление лексикографического порядка на множестве решений. Для этого упорядочим объекты множества I от 1 до $|I|$.

Среди множества эквивалентных решений выберем одно следующим образом. В каждой группе найдем объект с наименьшим номером. Упорядочим группы по возрастанию этих номеров и пронумеруем группы от 1 до p согласно полученному порядку. Построенное таким образом решение будем называть *лексикографически минимальным решением* среди эквивалентных ему решений.

Например, разобьем множество $\{1, \ldots, 10\}$ на группы $\{4, 6, 8, 9\}$, $\{1, 2, 7\}$, $\{3, 5, 10\}$. Согласно установленному выше лексикографическому порядку группа $\{1, 2, 7\}$ будет первой, $\{3, 5, 10\}$ — второй, $\{4, 6, 8, 9\}$ — третьей.

Разберем теперь, как шаг за шагом, пользуясь описанными уже правилами 1 и 2, построить линейные ограничения, реализующие предложенный подход.

Назначения в группу 1. Объект 1 должен лежать в группе 1, т. е. $x_{11} = 1$. Заметим, что остальные переменные x_{1k}, при $k > 1$, равны нулю.

Назначения в группу 2. Если объект 2 не лежит в группе 1, то он должен лежать в группе 2, т. е.

$$\text{если } x_{21} = 0, \text{ то } x_{22} = 1.$$

Получаем неравенство:

$$(1 - x_{22}) \leq x_{21}.$$

Если объект 2 лежит в группе 1 и объект 3 не лежит в группе 1, то он должен лежать в группе 2, т. е.

$$\text{если } x_{21} = 1 \text{ и } x_{31} = 0, \text{ то } x_{32} = 1.$$

Получаем неравенство:

$$(1 - x_{32}) \leq (1 - x_{21}) + x_{31}.$$

Далее если объекты $2, \ldots, (j-1)$ лежат в группе 1 и объект j не лежит в группе 1, тогда объект j должен быть в группе 2, т. е.

$$\text{если } x_{i1} = 1 \text{ для всех } i = 2, \ldots, (j-1) \text{ и } x_{j1} = 0, \text{ то } x_{j2} = 1.$$

Получаем неравенство:

$$(1 - x_{j2}) \leq \sum_{i=2}^{(j-1)} (1 - x_{i1}) + x_{j1}.$$

Назначения в группы с 3 по $(p-1)$. В каждую группу k $(k = 3, \ldots, p-1)$ должен попасть объект с наименьшими номером, который еще не попал в предыдущую группу с номерами от 1 до $(k-1)$. Значит, для любого объекта с номером j, если все предыдущие объекты с номерами от 2 до $(j-1)$ лежат в группе l, $l < k$, а j не лежит в l, то j должен быть в группе k.

Вспомним, что каждый объект принадлежит только одной группе, значит i может лежать в группе с меньшим номером l (меньшим, чем k) тогда и только тогда, когда $\sum_{l=1}^{k-1} x_{il} = 1$.

Таким образом, чтобы еще не размещенный ни в одной группе объект j определял следующий номер группы с 3 по $(p-1)$, должно быть выполнено условие: *если* $\sum_{l=1}^{(k-1)} x_{il} = 1$ *и* $\sum_{l=1}^{(k-1)} x_{jl} = 0$ *для всех* $i = 2, \ldots, j-1$, *то* $x_{jk} = 1$.

Следуя правилам 1 и 2, смоделируем эту импликацию в виде следующих неравенств:

$$(1 - x_{jk}) \leq \sum_{i=2}^{j-1} (1 - \sum_{l=1}^{k-1} x_{il}) + \sum_{l=1}^{k-1} x_{jl}, j = k, \ldots, n, k = 3, \ldots, (p-1)$$

или после приведения подобных:

$$\sum_{i=2}^{j-1} \sum_{l=1}^{k-1} x_{il} - \sum_{l=1}^{k} x_{jl} \leq j - 3, j = k, \ldots, n, k = 3, \ldots, (p-1).$$

Заметим, что переменные x_{ik} с индексами $i < k$ не нужны, так как согласно лексикографическому порядку никакой объект не может быть размещен в группе с большим, чем у самого объекта, номером.

Назначения в группу с номером p. Нет необходимости в том, чтобы выписывать условия для последней группы с номером p, поскольку все объекты, не размещенные в группы с меньшими номерами, автоматически будут назначены в последнюю.

3. Примеры математических моделей целочисленного линейного программирования

В задачах *комбинаторной* оптимизации оптимальное решение требуется выбрать из конечного множества возможных решений. Этот раздел содержит ряд примеров задач комбинаторной оптимизации, для которых можно записать математические модели с целочисленными неотрицательными переменными и линейными связями между ними. В частности, для одного из основных представителей этого класса, задачи коммивояжера, будет описано несколько разных моделей, отличающихся количеством переменных и ограничений. Будет представлена модель целочисленного линейного программирования для задачи о двухстадийном гильотинном раскрое материала. Также будут рассмотрены примеры для хорошо известных в дискретной оптимизации задач о потоке и планировании производства интересные с точки зрения приемов моделирования.

3.1. Задача о потоке минимальной стоимости

В общем виде задача формулируется следующим образом. Задана сеть — ориентированный граф с множеством вершин V, в которых могут размещаться предприятия (поставщики, производители и пр.) некоторой продукции, множеством дуг A и двумя специальными вершинами: источником и стоком. Дуга (i, j) означает, что из вершины i в вершину j может осуществляться поставка некоторой продукции. Каждая дуга имеет неотрицательный вес u_{ij}, соответствующий пропускной способности этой дуги. Известны c_{ij} — удельные стоимости перевозки продукции вдоль дуги (i, j). Каждая вершина i имеет вес b_i. В зависимости от знака величины b_i вес вершины может означать:

– спрос на продукцию в этой вершине (или сколько продукции должно быть доставлено в вершину i), если $b_i < 0$;

– предложение продукции в этой вершине (или сколько продукции можно вывести из вершины i), если $b_i > 0$;

– транзитная вершина (продукция не остается и не забирается из вершины i), если $b_i = 0$.

Предполагается, что $\sum_{i \in V} b_i = 0$.

Задача заключается в том, чтобы доставить в каждую вершину нужное количество продукции из вершин, в которых она находится, с минимальными затратами, не нарушая пропускных возможностей дуг.

Введем неотрицательные переменные x_{ij} — величина потока, пропускаемого по дуге (i, j).

Целевая функция — минимальные суммарные затраты на перевозки внутри сети:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \tag{3.1}$$

при ограничениях на пропускные способности для каждой дуги:

$$x_{ij} \le u_{ij}, \quad (i,j) \in A, \tag{3.2}$$

для каждой вершины i должно выполняться *условие баланса* между потоком, входящим в эту вершину, потоком, выходящим из нее, и количеством продукции, доставляемой или вывозимой из этой вершины:

$$\sum_{j \in V} x_{ji} - \sum_{j \in V} x_{ij} = b_i, \quad i \in V, \tag{3.3}$$

и ограничениях на значения переменных:

$$x_{ij} \ge 0, \quad (i,j) \in A. \tag{3.4}$$

Полученная модель линейного программирования (3.1)–(3.4) соответствует *задаче о потоке минимальной стоимости.*

В некоторых задачах за осуществление перевозок, кроме удельных затрат, взимаются затраты, которые не зависят от объемов перевозок. Пусть h_{ij} — плата за использование дуги (i,j). В этом случае нужны дополнительные булевы переменные:

$$y_{ij} = \begin{cases} 1, & \text{если по дуге } (i,j) \text{ выполняются перевозки,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Целевая функция изменится, в ней появятся дополнительные слагаемые:

$$\min \sum_{(i,j) \in A} (c_{ij} x_{ij} + h_{ij} y_{ij}). \tag{3.5}$$

Переменные x_{ij} и y_{ij} связаны следующим образом:

$$\text{если } y_{ij} = 0, \text{ то } x_{ij} = 0, \text{ причем } 0 \le x_{ij} \le u_{ij}.$$

Согласно следствию из первого правила (разд. 2.2) эта зависимость моделируется следующим образом:

$$x_{ij} \le u_{ij} y_{ij}, \quad (i,j) \in A. \tag{3.6}$$

Модель (3.3)–(3.6) с $y_{ij} \in \{0,1\}$, для всех $(i,j) \in A$, является моделью смешанного целочисленного программирования и соответствует *задаче о потоке с фиксированными затратами* [30].

Подобные задачи о потоках возникают при моделировании систем подачи воды, отопления. Этот класс задач обладает важным свойством: если все веса b_i и пропускные способности u_{ij} являются целыми числами, то оптимальное решение будет целочисленным. Это свойство задачи будет важным при обсуждении вопроса о разрыве целочисленности в разд. 4.

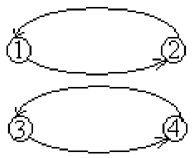

Рис. 6. Несколько циклов

3.2. Задача коммивояжера

Дано n городов. Известно c_{ij} — время перемещения из города i в город j. Коммивояжер выезжает из какого-то города, объезжает все города и возвращается в исходный. Необходимо найти такую последовательность посещения городов, при которой суммарное время перемещения было бы минимальным.

Введем n^2 булевых переменных x_{ij}, $i = 1, \dots, n$, $j = 1, \dots, n$:

$$x_{ij} = \begin{cases} 1, & \text{если коммивояжер едет из города } i \\ & \text{в город } j, \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Тогда суммарное время передвижения составляет

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}.$$

Однократный выезд из города i задается условием

$$\sum_{j=1}^{n} x_{ij} = 1, \tag{3.7}$$

однократный въезд в город i задается условием

$$\sum_{j=1}^{n} x_{ji} = 1. \tag{3.8}$$

Однако этих условий недостаточно, так как может возникать несколько циклов. Например, при $n = 4$ решение со значениями переменных $x_{12} = x_{21} = x_{34} = x_{43} = 1$ удовлетворяет ограничениям (3.7), (3.8), но не является решением задачи, так как образует два цикла, как показано на рис. 6.

В 1954 г. Данциг предложил следующий способ исключения такой ситуации. Пусть имеется два цикла. Чтобы получить из них обход всех городов, нужно, чтобы вместо одного ребра между парой вершин в цикле была пара ребер, соединяющая вершины, не попавшие в этот цикл. Значит, хотя бы для одной пары городов i и j, лежащих в разных циклах, переменная x_{ij} должна быть равна 1.

Обозначим через I множество всех городов. Пусть J — некоторое подмножество множества I, тогда условия, запрещающие всевозможные подциклы, выглядят следующим образом:

$$\sum_{i \in J} \sum_{j \in I \setminus J} x_{ij} \geq 1, \quad \forall J \subset I$$

или

$$\sum_{i \in J} \sum_{j \in J} x_{ij} \leq |J| - 1, \quad \forall J \subset I.$$

В результате получаем следующую математическую модель в терминах целочисленного линейного программирования:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{3.9}$$

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i \in I, \tag{3.10}$$

$$\sum_{j=1}^{n} x_{ji} = 1, \quad i \in I, \tag{3.11}$$

$$\sum_{i \in J} \sum_{j \in I \setminus J} x_{ij} \geq 1, \quad J \subset I, \quad 2 \leq |J| \leq |I| - 2, \tag{3.12}$$

$$x_{ij} \in \{0, 1\}, \quad i, j \in I. \tag{3.13}$$

Заметим, что задача о назначениях, рассмотренная в разд. 1, тесно связана с задачей коммивояжера. Максимальное количество подциклов в задаче о назначениях равно n, когда каждая машина i назначается на выполнение работы i. Минимальное количество циклов равно 1, и тогда такое назначение соответствует решению задачи о коммивояжере. Таким образом, множество допустимых решений в задаче о назначениях шире, чем множество допустимых решений в задаче о коммивояжере, поэтому оптимальное решение задачи о назначениях дает нижнюю оценку для оптимального решения задачи о коммивояжере. Покажем, что такая оценка может сколько угодно сильно отличаться от оптимального решения задачи коммивояжера.

Предположим, что оптимальное решение задачи о назначениях состоит из двух подциклов. Пусть элементы c_{ij}, участвующие в подциклах, равны p, а остальные элементы равны q, $q > p$ (рис. 7).

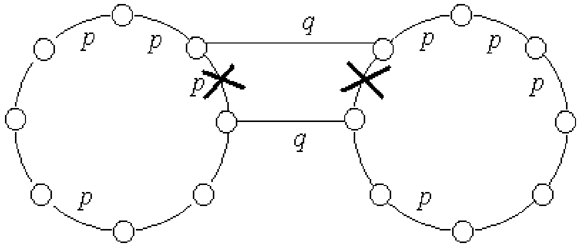

Рис. 7. Ликвидация подциклов в задаче коммивояжера

Чтобы получить оптимальное решение задачи о коммивояжере, т. е. один цикл, необходимо «разорвать» первый и второй подциклы, как показано на рис. 7, т. е. удалить два ребра длиной $2p$ и добавить два ребра длиной $2q$. Если в первом подцикле k ребер, а во втором — s, то значение целевой функции в задаче о назначениях на оптимальном решении равно $(k+s)p$. Для оптимального решения задачи о коммивояжере получаем значение целевой функции, равное $(n-2)p+2q$. Тогда разница между оптимальными значениями составляет $2(q-p)$. Следовательно, при фиксированном значении p величина q может быть выбрана так, что разность $2(q-p)$ может быть сколько угодно большой.

Недостатком приведенной модели является то, что число ограничений экспоненциально, их $(n+n+(2^{n-1}-n-1))$. В работе Таккера в 1960 г. была предложена постановка с полиномиальным числом ограничений.

Рассмотрим n целочисленных переменных u_i, соответствующих номеру шага, на котором посетили город i, тогда ограничения

$$u_i - u_j + nx_{ij} \le n-1, \quad i,j = 2,\ldots,n, \quad i \ne j \tag{3.14}$$

запрещают подциклы. Проверим, что эти условия действительно гарантируют наличие ровно одного цикла, проходящего через все города.

Пусть имеется 2 цикла. Один из них не проходит через первый город, обозначим его $(i_2,\ldots,i_k,i_2)$. Выпишем для каждой пары городов, идущих по порядку в этом цикле, условия (3.14):

$$u_{i_2} - u_{i_3} + n \le n-1,$$

$$u_{i_3} - u_{i_4} + n \le n-1,$$

$$\vdots$$

$$u_{i_k} - u_{i_2} + n \leq n - 1.$$

Сложив эти неравенства, получим $kn \leq (n-1)k$, что невозможно при $k \neq 0$.

Проверим, что цикл, проходящий через все города, удовлетворяет условиям (3.14), т. е. можно подобрать соответствующие значения переменных u_i. Пусть $u_i = k$, если город i посетили на шаге k, тогда неравенство $u_i - u_j \leq n-1$ верно при $x_{ij} = 0$. Если $x_{ij} = 1$, то $u_i = k$, а $u_j = (k+1)$, тогда $k-(k+1)+n \leq n-1$, $n-1 \leq n-1$.

Итак, (3.9)—(3.11), (3.13), (3.14), $u_i \geq 0, \quad i \in I$ — математическая модель задачи коммивояжера с полиномиальным числом ограничений.

Несмотря на то, что в этой модели существенно меньше переменных, она обладает недостатком, о котором речь пойдет в разд. 4.

Рассмотрим еще одну постановку задачи коммивояжера с полиномиальным числом переменных и ограничений, основанную на идеях моделирования задачи о потоке.

Оставим прежние булевы переменные x_{ij} и введем полиномиальное количество новых переменных: y_{ij} — поток некоторой продукции между городами i и j, тогда ограничения (3.10), (3.11) и следующий набор ограничений представляют все гамильтоновы циклы в задаче о коммивояжере:

$$y_{ij} - (n-1)x_{ij} \leq 0, \quad i,j \in I, \tag{3.15}$$

$$\sum_{j \in I} y_{1j} = n-1, \tag{3.16}$$

$$\sum_{j \in I} y_{ij} - \sum_{k \in I} y_{ki} = -1, \text{ для всех } i \in I, i \neq 1, \tag{3.17}$$

$$y_{ij} \geq 0, \quad i,j \in I, \tag{3.18}$$

$$x_{ij} \in \{0,1\}, \quad i,j \in I. \tag{3.19}$$

Ограничения (3.15) гарантируют перевозку продукции между городами i и j, только если $x_{ij} = 1$. Ограничение (3.16) говорит о том, что из первого города нужно развести $(n-1)$ единиц товара по остальным городам, оставляя, согласно (3.17), в каждом городе ровно одну единицу товара. Чтобы развести всю продукцию, сеть должна быть связанной, таким образом, подциклов в обходе возникать не будет. Итак, в этой модели (n^2+3n) ограничений и $2n^2$ переменных.

3.3. Задача о покрытии

Задано n возможных мест расположения пожарных станций. Величины c_j, $j = 1, \ldots, n$ задают стоимость размещения пожарной станции в месте j. Известно m объектов, на которые должны выезжать пожарные бригады. Причем пожарной бригаде имеет смысл выезжать на объект, если он находится на расстоянии не более D км, поэтому все объекты разбиты на группы объектов M_j, которые можно достичь из пожарной станции j.

Нужно определить, в каких местах разместить пожарные станции, чтобы все объекты были достижимы из действующих пожарных станций, а суммарные затраты на их размещение были бы минимальными.

Построим вспомогательную матрицу $(a_{ij})_{m\times n}$, чтобы задать разбиение объектов на группы:

$$a_{ij} = \begin{cases} 1, & \text{если объект } i \text{ попадает в группу } M_j, \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Введем n булевых переменных x_j, $j = 1, \dots, n$:

$$x_j = \begin{cases} 1, & \text{если в пункте } j \text{ размещается пожарная станция,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Целевая функция задачи — минимальные суммарные затраты на открытие станций:

$$\min \sum_{j=1}^{n} c_j x_j \tag{3.20}$$

при ограничениях, что каждый объект должен быть достижим хотя бы из одной пожарной станции:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1, \quad i = 1, \dots, m \tag{3.21}$$

и ограничения на значения переменных:

$$x_j \in \{0, 1\}, \quad j = 1, \dots, n. \tag{3.22}$$

Модели вида (3.20)–(3.22) соответствуют *задачам о покрытии множества.*

В рассмотренной задаче допускалось, что один объект может обслуживаться несколькими пожарными станциями, неравенство (3.21) гарантировало выполнение этого условия. Заменим в (3.21) знаки неравенства на равенства:

$$\sum_{j=1}^{n} a_{ij} x_j = 1, \quad i = 1, \dots, m. \tag{3.23}$$

Получим более жесткое требование, которое означает, что нужно построить разбиение множества объектов на непересекающиеся подмножества. Модели вида (3.20), (3.22), (3.23) соответствуют задачам о разбиении множества на непересекающиеся подмножества. Заметим, что решение задачи о разбиении является также решением задачи о покрытии, но обратное неверно. Задача о разбиении может не иметь решения при разрешимой задачи о покрытии. Например, пусть $n = 3$, $m = 3$, $M_1 = \{1, 2\}$, $M_2 = \{1, 3\}$, $M_3 = \{2, 3\}$. Тогда задача о покрытии имеет несколько решений: $x^1 = (1, 0, 1)$, $x^2 = (0, 1, 1)$, а задача о разбиении не разрешима. Если же $M_3 = \{3\}$, то задача о разбиении имеет решение: $x = (1, 0, 1)$.

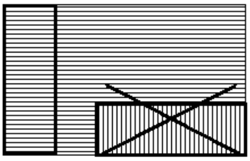

Рис. 8. Расположение предметов без поворотов

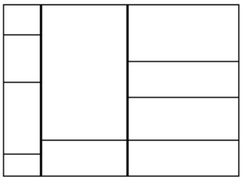

Рис. 9. Двухстадийный раскрой

3.4. Задача о двухстадийном гильотинном раскрое

Задачи о раскрое имеют широкое применение в различных отраслях промышленности: машиностроении, деревообработке, стеклопроизводстве [29]. Их суть в том, чтобы разместить и вырезать заготовки из материала с учетом различных технологических требований. Задачи классифицируются по форме заготовок, их размерности, способу разрезов. Часто по технологии можно выполнять только *гильотинные разрезы*, когда каждый разрез листа выполняется перпендикулярно одной из сторон листа и параллельно другой стороне от одного края до другого. В зависимости от узора на поверхности определяется ориентация вырезаемых кусков. Например, если на кусках задан рисунок, то повороты могут быть запрещены (рис. 8).

В *двухстадийной задаче* разрезы выполняются постадийно. На первой стадии лист разрезается по вертикали на несколько параллельных полос, на второй стадии каждая полоса разрезается по горизонтали на прямоугольные куски (рис. 9).

В результате многие прямоугольники могут оказаться большего размера, чем нужно, тогда выполняется завершающая стадия по обрезанию остатков материала. На рис. 10 приводится пример двухстадийного раскроя материала с последующим обрезанием остатков.

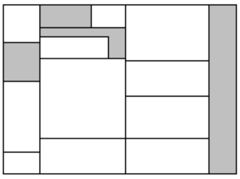

Рис. 10. Двухстадийный раскрой с подрезкой остатков материала

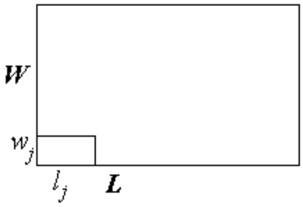

Рис. 11. Линейные размеры для задачи о раскрое

С точки зрения оптимизации можно сформулировать различные задачи. Например, минимизировать число листов, из которых необходимо вырезать заданное количество кусков.

Задача о двухстадийном гильотинном раскрое

Имеется один лист материала шириной W, длиной L и n типов прямоугольных заготовок шириной w_j и длиной l_j каждая, $j = 1, \ldots, n$ (рис. 11). Каждую заготовку можно вырезать в любом количестве. Известен доход p_j от продажи заготовки j. Требуется определить, какие заготовки и в каком количестве нужно вырезать, чтобы получить максимальный доход. Повороты прямоугольников при выкраивании запрещены, раскрой листа — гильотинный.

Поскольку раскрой гильотинный, значит на первой стадии лист разрезается по вертикали. Пусть k — максимальное число вертикальных полос, тогда k можно оценить сверху величиной $\lfloor \frac{L}{l_{min}} \rfloor$, где l_{min} — длина самого короткого прямоугольника, $l_{min} = min_{j \in \{1,\ldots,n\}}\{l_j\}$. Пусть m_j — максимальное количество прямоугольников j-го типа, которое можно выкроить из данного листа, m_j можно оценить сверху величиной $\lfloor \frac{LW}{l_j w_j} \rfloor$.

Введем следующие целочисленные переменные:

y_i — длина i-й полосы, $i = 1, \dots, k$,

x_{ij} — количество прямоугольников j-го типа, выкраиваемых из i-й полосы.

Чтобы выписать условие на вместимость прямоугольника j-го типа в i-ю полосу, нужно ввести вспомогательные булевы переменные:

$$x'_{ij} = \begin{cases} 1, & \text{если прямоугольник } j\text{-го типа} \\ & \text{выкраивается из } i\text{-й полосы,} \\ 0 \; - & \text{в противном случае.} \end{cases}$$

Используя введенные переменные, выпишем математическую модель.

Целевая функция задачи — максимальный суммарный доход:

$$\max \sum_{i=1}^{k} \sum_{j=1}^{n} p_j x_{ij},$$

ограничения задачи:

– общая ширина всех заготовок, выкраиваемых из i-й полосы, не превышает ширины листа

$$\sum_{j=1}^{n} w_j x_{ij} \leq W, \quad i = 1, \dots, k;$$

– суммарная длина всех полос не превышает длины листа:

$$\sum_{i=1}^{k} y_i \leq L;$$

– длина i-й полосы позволяет выкроить j-ю заготовку:

$$l_j x'_{ij} \leq y_i, \quad i = 1, \dots, k, j = 1, \dots, n.$$

Необходима еще группа ограничений, чтобы связать булевы переменные x'_{ij} с целочисленными переменными x_{ij} и гарантировать, что если j-я заготовка выкраивается из i-й полосы, то можно выкроить ограниченное число заготовок, и наоборот, нельзя выкраивать j-ю заготовку из i-й полосы, если она не помещается по размерам, т. е. нужно смоделировать два взаимоисключающих условия:

$$\text{если } x'_{ij} = 0, \text{то } x_{ij} = 0,$$

$$\text{если } x'_{ij} = 1, \text{то } x_{ij} \leq m_j.$$

Это можно реализовать с помощью следующих неравенств:

$$x_{ij} \leq m_j x'_{ij}, \quad i = 1, \dots, k, j = 1, \dots, n.$$

Последняя группа ограничений на значения переменных:

$$x_{ij} \leq m_j, \quad i = 1, \dots, k, j = 1, \dots, n,$$

$$x_{ij} \geq 0, \text{ целые}, \ i = 1, \ldots, k, j = 1, \ldots, n,$$
$$x'_{ij} \in \{0, 1\}, \quad i = 1, \ldots, k, j = 1, \ldots, n,$$
$$y_i \in \{1, 2, \ldots, L\}, \quad i = 1, \ldots, k.$$

3.5. Задача о разрезе балок

В деревообрабатывающий цех поступил заказ вырезать короткие бруски прямоугольной формы из длинных прямоугольных балок. Высота балок и брусков одинаковая. Заказ состоит из b разных типов брусков, в количестве n_i штук каждого, $i = 1, \ldots, b$. На складе имеется l разных типов балок длиной L_j каждая в количестве Q_j, $j = 1, \ldots, l$. Бруски можно вырезать из имеющихся балок, в этом случае стоимость выполнения одного разреза составляет c у. е., можно купить готовые по цене p_i у. е. за один брусок i-го типа. Задача заключается в том, чтобы выполнить заказ на бруски с минимальными суммарными затратами.

Особенность этой задачи в том, что, прежде чем выписать математическую модель, нужно составить всевозможные схемы (варианты) раскроя балок или *карт раскроя*, учитывая длину балок L_j. Причем из всевозможных карт раскроя исключим нерациональные варианты, т. е. те схемы, которые порождают большие отходы. Под большими отходами подразумеваются такие излишки материала, из которых можно было бы выкроить бруски.

Пусть s_j — количество возможных вариантов раскроя j-й балки, k_{is}^j — количество брусков i-го типа, выкраиваемых из j-го типа балки по s-му варианту, f_s^j — столько разрезов выполняется на j-й балке по s-му варианту, где $j = 1, \ldots, l$ $i = 1, \ldots, b$ $s = 1, \ldots, s_j$. Обрезание остатков балки до нужной длины бруска тоже считается разрезом.

Введем неотрицательные целочисленные переменные y_s — столько раз используется s-я карта раскроя, $s = 1, \ldots, (\sum_{j=1}^{l} s_j)$, неотрицательные целочисленные переменные x_i — количество купленных готовых брусков, $i = 1, \ldots, b$.

Целевая функция — минимальные суммарные затраты, которые складываются из затрат на разрезание и закупку:

$$\min \sum_{j=1}^{l} \sum_{s=1}^{s_j} c f_s^j y_s + \sum_{i=1}^{b} p_i x_i.$$

Общее число балок разной длины не превосходит имеющегося количества на складе:

$$\sum_{s=1}^{s_j} y_s \leq Q_j, \quad j = 1, \ldots, l.$$

Необходимо выполнить заказ на бруски разной длины:

$$\sum_{j=1}^{l} \sum_{s=1}^{s_j} k_{is}^j y_s + x_i \geq n_i, \quad i = 1, \ldots, b.$$

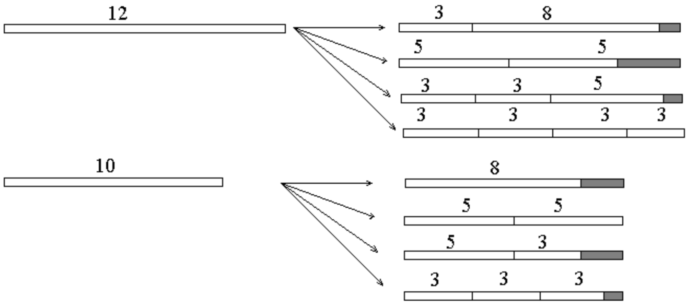

Рис. 12. Возможные схемы раскроя балок

Допустимые значения переменных:

$$y_s \geq 0, \text{ целые}, \quad s = 1, \dots, s_j, \quad j = 1, \dots, l,$$

$$x_i \geq 0, \text{ целые}, \quad i = 1, \dots, b.$$

Заметим, что эта модель обладает тем недостатком, что количество схем раскроя определяет количество переменных в задаче, т. е. число строк или столбцов в матрице ограничений. Количество схем раскроя конечное, но может оказаться колоссально большим, в связи с чем возникают вычислительные трудности у универсальных точных методов. Однако их можно преодолеть, пользуясь *техникой генерации столбцов* [12], которая решает задачу, не просматривая все столбцы или строки.

Пример

Заказ состоит из 60 брусков длиной 8 м, 40 брусков длиной 5 м и 75 брусков длиной 3 м. Длина балок составляет 12 и 10 м, в наличии имеется 20 и 25 штук соответственно. Стоимость выполнения одного разреза составляет 0,5 у.е., готовые бруски покупаются по цене 2 у.е., 1,5 у.е., 1,1 у.е. за восьми-, пяти- и трехметровый брусок соответственно. Выполнить заказ на бруски с минимальными суммарными затратами.

В данном случае получается 8 карт раскроя (рис. 12). Математическая модель выглядит следующим образом:

$$\min y_1 + y_2 + 1.5y_3 + 1.5y_4 + 0.5y_5 + 0.5y_6 + y_7 + 1.5y_8 + 2v_1 + 1.5v_2 + 1.1v_3$$

$$y_1 + y_2 + y_3 + y_4 \leq 20,$$

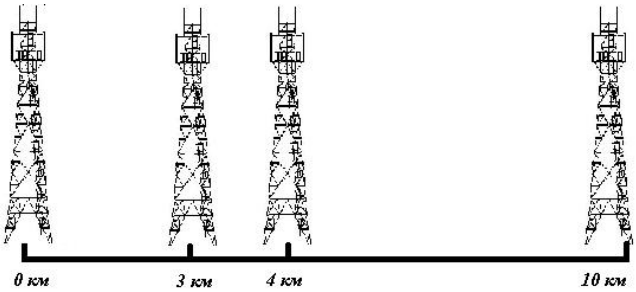

Рис. 13. Расположение башен

$$y_5 + y_6 + y_7 + y_8 \leq 25,$$
$$y_1 + y_5 + v_1 \geq 60,$$
$$2y_2 + y_3 + 2y_6 + y_7 + v_2 \geq 40,$$
$$y_1 + 2y_3 + 4y_4 + y_7 + 3y_8 + v_3 \geq 75,$$
$$y_i \geq 0, \text{ целые}, \quad i = 1, \ldots, 8,$$
$$v_j \geq 0, \text{ целые}, \quad i = 1, \ldots, 3.$$

3.6. Задача о башнях

Имеется конечное множество клиентов $\{1, \ldots, n\}$ и конечное множество возможных мест, где можно размещать башни сотовой связи $\{1, \ldots, m\}$. Известны расстояния между клиентами и башнями. Каждый клиент прикрепляется к ближайшей башне, если ближайших башен несколько, то к любой из них. Нужно разместить ровно p башен так, чтобы после прикрепления клиентов к башням разница между максимальным и минимальным числом клиентов, закрепленных за башнями, была минимальной.

Например, пусть башни располагаются в четырех населенных пунктах вдоль дороги: в начале дороги, через 3 км, через 4 км и через 10 км от начала дороги. Клиенты расположены в этих же пунктах (рис. 13). Нужно разместить ровно 2 башни. В табл. 5 приводится шесть возможных вариантов размещения и соответствующее значение целевой функции.

Таблица 5

Возможные варианты размещения башен

Номера активных башен	Номер башни, обслуживающей соответствующий пункт				Разница в количестве клиентов
	1	2	3	4	
(1,2)	1	2	2	2	2
(1,3)	1	3	3	3	2
(1,4)	1	1	1	4	2
(2,3)	2	2	3	3	0
(2,4)	2	2	2	4	2
(3,4)	3	3	3	4	2

Сформулируем эту задачу в виде задачи целочисленного линейного программирования. Введем булевы переменные:

$$y_j = \begin{cases} 1, & \text{если в } j\text{-м пункте размещается башня,} \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-й клиент обслуживается башней в } j\text{-м пункте,} \\ 0 \ - & \text{в противном случае,} \end{cases}$$

и неотрицательные целочисленные переменные:
u — максимальное число клиентов, обслуживаемых одной башней,
l — минимальное число клиентов, обслуживаемых одной башней.

Тогда в целевой функции минимизируется разница:

$$\min(u - l), \tag{3.24}$$

при условиях:

$$x_{ij} \le y_j, \quad i \in \{1, \dots, n\}, j \in \{1, \dots, m\}, \tag{3.25}$$

$$\sum_{j=1}^{m} y_j = p, \tag{3.26}$$

$$\sum_{j=1}^{m} x_{ij} = 1, \quad i \in \{1, \dots, n\}, \tag{3.27}$$

$$u \ge \sum_{i=1}^{n} x_{ij}, \quad j \in \{1, \dots, m\}, \tag{3.28}$$

$$l \le \sum_{i=1}^{n} x_{ij} + n(1 - y_j), \quad j \in \{1, \dots, m\}, \tag{3.29}$$

$$\sum_{l=1}^{m} c_{il}x_{il} + (M_i - c_{ij})y_j \leq M_i, \quad i \in \{1, \dots, n\}, j \in \{1, \dots, m\}, \tag{3.30}$$

и ограничениях на допустимые значения переменных:

$$x_{ij} \in \{0, 1\}, \quad i \in \{1, \dots, n\}, j \in \{1, \dots, m\}, \tag{3.31}$$

$$y_j \in \{0, 1\}, \quad j \in \{1, \dots, m\}, \tag{3.32}$$

$$u, l \in \mathbb{Z}^+, \tag{3.33}$$

где $M_i = \max_{j \in \{1,\dots,m\}} c_{ij}$, $i \in \{1, \dots, n\}$.

Группа ограничений (3.25) уже встречалась в моделях размещения, она гарантирует, что обслуживание клиентов будет только из «активных» башен. Ограничение (3.26) насчитывает число размещенных башен. Ограничения (3.27) гарантируют, что каждый клиент обслуживается одной башней. Ограничения (3.28) и (3.29) определяют максимальное и минимальное число клиентов по всем башням. Для того, чтобы переменная l не обращалась в нуль, в правой части ограничений (3.29) добавлено слагаемое $n(1-y_j)$, которое позволяет при $y_j = 0$ не ограничивать значение переменной l сверху, а при $y_j = 1$ насчитать минимальное число клиентов и выбрать башню с наименьшей загруженностью. Ограничения (3.30) гарантируют, что каждому клиенту соответствует ближайшая башня.

4. Анализ качества моделей целочисленного линейного программирования

4.1. Классификация моделей

В предыдущих разделах были описаны практические правила и идеи о том, как записать математическую модель в терминах *смешанного целочисленного линейного программирования*, рассмотрены разные примеры практического характера. В этом разделе продолжается рассмотрение моделей смешанного целочисленного линейного программирования, но уже с другой точки зрения. Здесь приводятся формальные определения, разные эквивалентные формы записи задач смешанного целочисленного линейного программирования в общем виде и теоретические аспекты оценки качества моделей. Обсуждается, как практически выбрать и построить наилучшую формулировку задачи, улучшить модель, чтобы сократить разрыв целочисленности.

Математическая модель следующего вида называется *моделью смешанного целочисленного линейного программирования, записанная в стандартной форме* (сокращенно MIP, от англ. mixed integer programming):

$$\min(cx + fy) \tag{4.1}$$

$$Ax + By \geq b, \tag{4.2}$$

$$x \geq 0, \tag{4.3}$$

$$y \geq 0, \text{ целые}, \tag{4.4}$$

где $c = (c_1, \ldots, c_n)$, $f = (f_1, \ldots, f_p)$ — вектор-строки с вещественными компонентами, задающие коэффициенты целевой функции;
$x = (x_1, \ldots, x_n)^T$ — вектор-столбец переменных задачи, принимающих неотрицательные вещественные значения;
$y = (y_1, \ldots, y_p)^T$ — вектор-столбец переменных задачи, принимающих неотрицательные целочисленные значения;
A, B – матрицы размерности $(m \times n)$ и $(m \times p)$ соответственно, с рациональными значениями компонент, определяющие коэффициенты в системе из m ограничений;
$b = (b_1, \ldots, b_m)^T$ — вектор-столбец с вещественными компонентами правых частей ограничений.

Обозначим через X_{MIP} множество значений переменных, удовлетворяющих ограничениям (4.2)–(4.4). Множество X_{MIP} называют *множеством допустимых* решений задачи. Обозначим через $z^*(X_{MIP})$ оптимальное значение целевой функции (4.1) на множестве X_{MIP}.

Заметим, что в стандартной форме требуется, чтобы задача была на минимум, все знаки в ограничениях были нестрогими, вида $\geq$, а переменные принимали неотрицательные значения.

Существуют каноническая и общая формы записи MIP. Каждую из этих форм можно преобразовать в стандартную, пользуясь следующими простыми соображениями:

— ограничения равенства $Ax + By = b$ эквивалентны паре неравенств $Ax + By \geq b$ и $-Ax - By \geq -b$;

— задача на максимум с целевой функцией $(cx+fy)$ эквивалентна задаче на минимум с целевой функцией $-(cx + fy)$. Аналогичные преобразования, т. е. умножение на (-1), нужно сделать в неравенствах со знаком $\leq$;

— каждое вхождение *свободной переменной* x, т. е. переменной без ограничения на знак, нужно заменить разностью двух новых неотрицательных переменных, $x = (u - v)$. Таким образом, все переменные в модели будут неотрицательными.

Любое оптимальное решение, полученное для модели в стандартной форме, нетрудно преобразовать в оптимальное решение исходной задачи, и наоборот.

Задача MIP, в которой все переменные принимают значения только 0 или 1, называют *задачей булева, или 0–1, программирования.*

Задача MIP, в которой все целочисленные переменные принимают значения 0 или 1, называют *задачей смешанного 0–1 программирования.*

Задача MIP, в которой все переменные принимают вещественные значения, называют *задачей линейного программирования* (сокращенно LP, от англ. linear programming).

Задача MIP, в которой все переменные принимают только целые значения, называют *задачей полностью целочисленного программирования* (сокращенно IP, от англ. integer programming).

Известно, что задача IP является NP-трудной в общем случае [5], но полиномиально разрешима при фиксированном числе переменных [22]. Задача IP имеет широкие рамки, внутри которых формулируются задачи MIP смешанного 0–1 программирования. Слабое место этих задач в том, что в общем виде не известно алгоритма, практически применимого для решения задач большой размерности.

Решать задачу LP в общем случае проще, чем соответствующую задачу MIP, потому что для LP существуют точные полиномиальные алгоритмы [13]. Поэтому иногда в задачах MIP отказываются от условий целочисленности и решают соответствующую задачу LP. Некоторые задачи обладают тем свойством, что оптимальное решение задачи LP является целочисленным, однако это не всегда так. Отказ от условий целочисленности в задаче MIP и переход к соответствующей задаче LP называют *линейной релаксацией.* Под линейной релаксацией задачи (4.1)–(4.4) понимается следующая задача:

$$\min(cx + fy) \tag{4.5}$$

$$Ax + By \geq b, \tag{4.6}$$

$$x \geq 0, \tag{4.7}$$

$$y \geq 0. \tag{4.8}$$

Будем обозначать задачу (4.5)–(4.8) через LR (от англ. relaxation — ослабление), а множество допустимых решений (4.6)–(4.8) через X_{LR}. Нетрудно заметить, что множество X_{LR} шире допустимого множества X_{MIP}: $X_{MIP} \subset X_{LR}$. Значит, оптимальное значение целевой функции задачи линейной релаксации дает нижнюю оценку на оптимум соответствующей задачи MIP: $z^*(X_{LR}) \le z^*(X_{MIP})$, если исходная задача на минимум, и дает верхнюю оценку, если исходная задача MIP на максимум.

4.2. Разрыв целочисленности

Величина отклонения оптимального значения линейной релаксации от оптимального значения целевой функции для задачи смешанного целочисленного программирования называется *разрывом целочисленности*. Рассматривают *абсолютный разрыв целочисленности*

$$|z^*(X_{LR}) - z^*(X_{MIP})|$$

и *относительный разрыв целочисленности*

$$\frac{|z^*(X_{LR}) - z^*(X_{MIP})|}{\max\{|z^*(X_{LR})|, |z^*(X_{MIP})|\}} \cdot 100\%.$$

Для вычисления разрыва нужно точно решить обе задачи. Часто вместо оптимальных значений рассматривают хорошие приближенные значения для $z^*(X_{LR})$ и $z^*(X_{MIP})$ и тогда вместо разрыва целочисленности вычисляют следующую величину:

$$\frac{UB - LB}{UB} \cdot 100\%,$$

где UB, LB — верхняя и нижняя оценка оптимального решения соответственно. Оптимальное значение должно лежать в промежутке $[LB, UB]$. Иногда в решателях разрыв вычисляется по отношению к нижней оценке, т. е. в знаменателе стоит LB вместо UB.

Рассмотрим несколько примеров, чтобы наглядно увидеть, к каким изменениям в оптимальном решении приводит добавление требований целочисленности переменных.

Пример [28]

Рассмотрим следующую задачу P_1:

$$\max(x_1 + x_2) \tag{4.9}$$

$$3x_1 + 5x_2 \le 15, \tag{4.10}$$

$$5x_1 + 2x_2 \le 10, \tag{4.11}$$

$$x_1 \ge 0, \tag{4.12}$$

$$x_2 \ge 0. \tag{4.13}$$

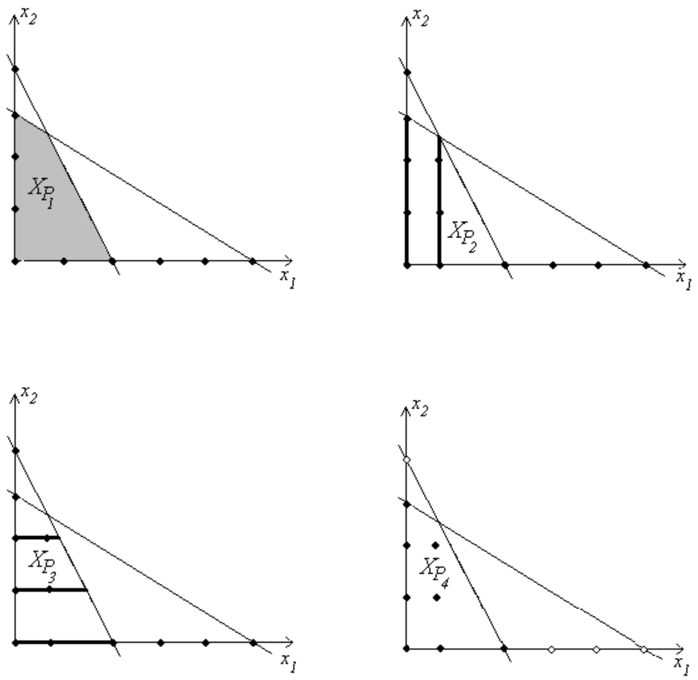

Рис. 14. Допустимые области для задач P_1, P_2, P_3, P_4

Заменим в задаче P_1 условие (4.12) на условие $x_1 \geq 0$, *целые* и обозначим новую задачу P_2. Заменим в задаче P_1 условие (4.13) на условие $x_2 \geq 0$, *целые* и обозначим новую задачу P_3. Заменим в задаче P_1 оба условия (4.12), (4.13) на $x_1 \geq 0$, *целые* и $x_2 \geq 0$, *целые*, полученную задачу IP обозначим P_4.

На рис. 14 изображено графическое представление допустимой области для каждой из задач P_1, P_2, P_3 и P_4.

Задача P_1 является задачей линейного программирования, и ее допустимая область X_{P_1} — это заштрихованная часть, в том числе и границы. Для задачи смешанного целочисленного программирования P_2 допустимая область X_{P_2} — это вертикальные отрезки, проходящие через точки с целочисленными значениями x_1, удовлетворяющие ограничениям задачи. Аналогично для задачи P_3 допустимая область X_{P_3} — это горизонтальные отрезки. Допустимая область X_{P_4} для задачи полностью целочисленного программирования P_4 состоит из восьми точек с целочисленными координатами, удовлетворяющих ограничениям задачи. Таким образом, после добавления требования целочисленности область X_{P_4} стала существенно меньше по сравнению с X_{P_1}, X_{P_2}, X_{P_3}. Рассмотрим оптимальные решения этих задач:

$$P_1 : z^*(X_{P_1}) = 3.4211 \quad x^* = (1.0526, 2.3684);$$

$$P_2 : z^*(X_{P_2}) = 3.4 \quad x^* = (1, 2.4);$$

$$P_3 : z^*(X_{P_3}) = 3.2 \quad x^* = (1.2, 2);$$

$$P_4 : z^*(X_{P_4}) = 3 \quad x^* = (1, 2).$$

Нетрудно заметить, что, переходя от решения задачи P_1 с наименее жесткими требованиями к задаче P_4 с более жесткими требованиями, оптимальное значение целевой функции уменьшается: $z^*(X_{P_4}) < z^*(X_{P_1})$.

Пример

В рассмотренных выше примерах оптимальное решение задачи P_4 может быть получено округлением решения задачи P_1 вниз. Однако округление не всегда приводит к оптимальному решению соответствующей задачи MIP. Рассмотрим следующую задачу:

$$\max(2x_1 + x_2) \tag{4.14}$$

$$7x_1 + 48x_2 \leq 84, \tag{4.15}$$

$$-x_1 + 12x_2 \geq 3, \tag{4.16}$$

$$x_1 \geq 0, \text{ целые}, \tag{4.17}$$

$$x_2 \geq 0, \text{ целые}. \tag{4.18}$$

Оптимальным решением соответствующей задачи линейного программирования будет $x^*_{LP} = (6.5455,\ 0.7955)$ со значением целевой функции $z^*_{LP} = 13.8864$. Оптимальным решением исходной задачи является вектор $x^*_{IP} = (5, 1)$ со значением целевой функции $z^*_{IP} = 11$. Из рис. 15 видно, что x^*_{IP} нельзя получить простым округлением решения x^*_{LP}.

Вычислим разрыв целочисленности. Абсолютный разрыв составляет $13.8864 - 11 = 2.8864$, относительный равен $\frac{2.8864}{13.8864} \cdot 100\ \% = 20.79\ \%$. Величина относительного разрыва дает информацию о том, насколько сложной в вычислительном плане окажется задача для точных методов, которые используют решение линейной релаксации при доказательстве оптимальности целочисленного решения. Обычно задачи с относительным разрывом 10 % считаются простыми, а задачи с разрывом 50 % — сложными. В связи с этим среди разных возможных математических моделей MIP лучше выбирать ту, у которой разрыв целочисленности меньше.

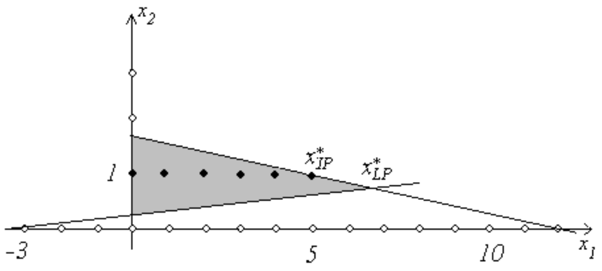

Рис. 15. Оптимальные целочисленное и дробное решения

Более того, может оказаться, что задача LP имеет решение, а соответствующая задача IP не разрешима. Например, заменим ограничение (4.16) на

$$-x_1 + 12x_2 \geq 13, \tag{4.19}$$

тогда область допустимых решений в задаче (4.14), (4.15), (4.17)–(4.19) — это треугольник ABC (рис. 16). Оптимальное решение соответствующей задачи LP находится в точке C. Заметим, что ни одна целочисленная точка не входит в область ABC, следовательно, множество допустимых решений в задаче IP (4.14), (4.15), (4.17)–(4.19) является пустым.

Пример

Пусть допустимая область образована неравенствами:

$$x_1 + x_2 \leq 10, \tag{4.20}$$

$$5x_1 + 3x_2 \geq 15, \tag{4.21}$$

$$x_1 \leq 6, \tag{4.22}$$

$$x_2 \leq 7, \tag{4.23}$$

$$x_1 \geq 0, x_2 \geq 0. \tag{4.24}$$

Если изобразить графически, то допустимая область — это многогранник $ABCDEF$ (рис. 17). Из теории линейного программирования известно, что оптимальное решение задачи LP лежит в одной из угловых точек допустимой области. В данном примере все угловые точки имеют целочисленные координаты. Следовательно, при любой целевой функции среди оптимальных решений задачи линейного программирования на этом множестве всегда будет целочисленное решение. В этом случае разрыв целочисленности равен нулю.

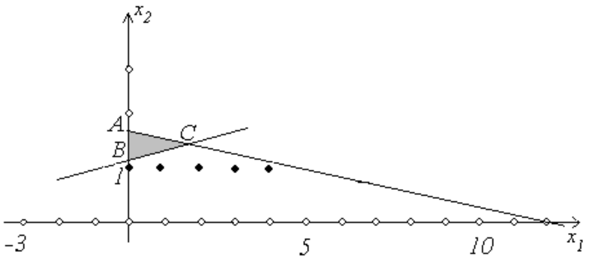

Рис. 16. Допустимая область — треугольник ABC

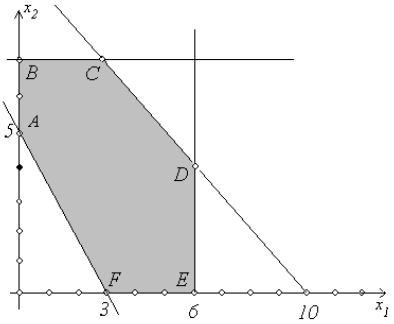

Рис. 17. Допустимая область — многогранник $ABCDEF$

Пример. Задача планирования производства

Завод занимается производством некоторой однотипной продукции. Продукция производится в начале месяца и либо отправляется потребителям сразу, либо хранится на складе. Продукцию можно производить и хранить на складе на протяжении всего горизонта планирования. Производственные затраты складываются из единовременных расходов на запуск производства, удельных производственных затрат и удельных затрат на хранение продукции. Задан промежуток времени T месяцев, в течение которого необходимо спланировать производство и хранение продукции так, чтобы выполнить заказ с минимальными затратами.

Обозначим через d_t — заказ на продукцию в месяц t, c_t — затраты на запуск производства в месяц t, p_t — удельные затраты на производство в месяц t, h_t — удельные затраты на хранение продукции в течение месяца t. Выпишем две модели с разными переменными.

Первая модель использует следующие переменные:

y_t — количество продукции, произведенной в месяц t;

s_t — количество продукции, оставленной на складе к концу месяца t;

$x_t = \begin{cases} 1, & \text{если в месяц } t \text{ было запущено производство,} \\ 0 \; - & \text{в противном случае.} \end{cases}$

Тогда получаем следующую математическую модель:

$$\min \sum_{t=1}^{T} (c_t x_t + p_t y_t + h_t s_t) \tag{4.25}$$

$$y_1 = d_1 + s_1, \tag{4.26}$$

$$s_{t-1} + y_t = d_t + s_t, \ t = 2, \ldots, T, \tag{4.27}$$

$$y_t \le \sum_{k=1}^{T} d_k x_t, \ t = 1, \ldots, T, \tag{4.28}$$

$$s_T = 0, \tag{4.29}$$

$$y_t \ge 0, \ s_t \ge 0, \text{ целые}, \ x_t \in \{0, 1\}, \ t = 1, \ldots, T. \tag{4.30}$$

Целевая функция (4.25) — минимальные общие затраты, которые складываются из затрат на запуск, производство и хранение. Ограничения (4.26) и (4.27) отвечают за удовлетворение спроса и выполнение «закона сохранения» продукции в каждом месяце. Поскольку в начальный момент времени на складе отсутствует продукция, то для первого месяца выписано отдельное условие. Ограничения (4.28) устанавливают связь между переменными x_t и y_t и ограничивают сверху максимальный ежемесячный объем партии. Только если производство запущено в месяц t, на нем может производится продукция, но не более, чем величина суммарного спроса на него. Ограничение (4.29) задает уровень запасов на складе в конце горизонта планирования.

Вторая модель использует следующие переменные: q_{it} — столько продукции производится в месяц i, чтобы выполнить заказ в месяц t, $t \geq i$, переменные x_t остались теми же, что и в предыдущей модели. Используя введенные переменные, получаем следующую математическую модель:

$$\min \sum_{t=1}^{T} \sum_{i=1}^{t} (p_i + h_i + h_{i+1} + \ldots + h_{t-1}) q_{it} + \sum_{t=1}^{T} c_t x_t, \tag{4.31}$$

$$\sum_{i=1}^{t} q_{it} = d_t, \ t = 1, \ldots, T, \tag{4.32}$$

$$q_{it} \leq d_t x_i, \ i = 1, \ldots, T, \ t = i, \ldots, T, \tag{4.33}$$

$$q_{it} \geq 0, \ x_t \in \{0, 1\}, \ t = 1, \ldots, T. \tag{4.34}$$

Вторая модель обладает тем свойством, что если требование целочисленности переменных x_t заменить на условие непрерывности переменных $0 \leq x_t \leq 1$, то в оптимальном решении соответствующей задачи линейной релаксации переменные x_t будут принимать значение 0 или 1, т. е. разрыв целочисленности для LR (4.31)–(4.34) равен нулю. Первая модель — это MIP, поскольку в ограничениях (4.28) имеются большие константы, то разрыв целочисленности не равен нулю.

Следующая теорема дает оценку близости решений IP и соответствующей LR.

Теорема [22]**.** *Пусть A — целочисленная $(m \times n)$-матрица, каждый минор которой по абсолютной величине не превосходит Δ, и пусть даны векторы b и c. Предположим, что оба максимума*

$$\max\{cx | Ax \leq b\} \tag{4.35}$$

и

$$\max\{cx | Ax \leq b, x \in \} \tag{4.36}$$

конечны. Тогда:

1. *Для любого оптимального решения y задачи (4.35) существует оптимальное решение z задачи (4.36), для которого $\|y - z\|_\infty \leq n\Delta$.*
2. *Для любого оптимального решения z задачи (4.36) существует оптимальное решение y задачи (4.35), для которого $\|y - z\|_\infty \leq n\Delta$.*

($\|y\|_\infty$ обозначает максимум абсолютных величин компонент вектора y.)

Эта теорема обобщает результат о том, что для любой рациональной матрицы A существует такой конечный набор векторов $x^1, \ldots, x^N$, что для любой вектор-строки c и любого целочисленного вектор-столбца b из того, что x^* является оптимальным базисным решением (вершиной) релаксированной задачи целочисленного линейного программирования $\max\{cx | x \geq 0, Ax = b\}$, следует, что оптимум IP $\max\{cx|x \geq 0, Ax = b, x \in \mathbb{Z}\}$ достигается на одном из векторов $(x^* - x^k)$, $k = 1, \ldots, N$.

Пример

Следующий пример показывает, что указанная в теореме граница $n\Delta$ не улучшаема.

Пусть

$$A := \begin{bmatrix} +1 & 0 & 0 & \cdots & 0 \\ -1 & +1 & 0 & \cdots & 0 \\ 0 & -1 & +1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & 0 & -1 & +1 \end{bmatrix},$$

$$b := \begin{bmatrix} \beta \\ \beta \\ \vdots \\ \beta \end{bmatrix},$$

$$c := (1, \ldots, 1)$$

для любого β, $0 \leq \beta < 1$ следующие векторы являются единственными решениями задач $\max\{cx|Ax \leq b\}$ и $\max\{cx|Ax \leq b, x \in \mathbb{Z}\}$:

$$y := \begin{bmatrix} \beta \\ 2\beta \\ \vdots \\ n\beta \end{bmatrix},$$

$$z := \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

и $\|y - z\|_\infty \leq n\Delta\beta$.

4.3. Число ограничений и переменных в модели

Качество построенной математической модели можно оценивать по разным показателям. Для задач булевого программирования одним из показателей является количество булевых переменных. Число переменных особенно важно при решении задач точными методами, одним из которых является метод ветвей и границ [13]. В процессе решения этим методом строится так называемое дерево ветвления. Число вершин в дереве зависит от количества булевых переменных. Таким образом, если в модели n булевых переменных, то в дереве ветвления может быть 2^n вершин в худшем случае. Следовательно, с ростом n время решения задачи может расти экспоненциально.

Рассмотрим на примерах несколько идей для сокращения количества переменных.

Пример

Имеется парк из трех грузовиков и два маршрута, по которым нужно отправить грузовики. Предполагается, что все грузовики одинаковые. Требуется назначить грузовики на маршруты. Это можно сделать несколькими способами.

Первый способ. Пусть

$$x_{ij} = \begin{cases} 1, & \text{если грузовик } i \text{ отправляется по маршруту } j, \\ 0 \ - & \text{в противном случае,} \end{cases}$$

где $i \in \{1, 2, 3\}$, $j \in \{1, 2\}$. Этот способ может привести к тому, что в дереве ветвления в худшем случае будет 2^6 вершин.

Второй способ. Пусть

$$y_i = \begin{cases} 1, & \text{если грузовик } i \text{ отправляется по первому маршруту,} \\ 0, & \text{если грузовик } i \text{ отправляется по второму маршруту,} \end{cases}$$

где $i \in \{1, 2, 3\}$. Этот способ может привести к тому, что в дереве ветвления в худшем случае будет 2^3 вершин.

Второй способ приводит к меньшему количеству переменных для ветвления. Однако оба способа обладают недостатком, обусловленным наличием симметрии в решениях, а именно: поскольку все грузовики одинаковые, то пара допустимых решений (первый способ)

$$x_{11} = 1, x_{22} = 1, x_{32} = 1 \tag{4.37}$$

и

$$x_{12} = 1, x_{21} = 1, x_{32} = 1, \tag{4.38}$$

и пара допустимых решений (второй способ)

$$y_1 = 1, y_2 = 0, y_3 = 0 \tag{4.39}$$

и

$$y_1 = 0, y_2 = 1, y_3 = 0 \tag{4.40}$$

по сути, представляют одно и то же назначение грузовиков на маршруты. Один из грузовиков обслуживает один маршрут, а два других грузовика обслуживают другой маршрут. В данном примере можно избежать симметрии и одновременно сократить число переменных введением других целочисленных переменных.

Третий способ. Пусть n_j — количество грузовиков, отправленных по маршруту j. Тогда решение

$$n_1 = 1, n_2 = 2$$

и решения (4.37), (4.38), (4.39) и (4.40), по сути, подразумевают одно и то же назначение грузовиков на маршруты, но теперь в модели, в три раза меньше переменных.

Однако в данном случае в модели нет булевых переменных, и процесс ветвления по целочисленным переменным в методе ветвей и границ будет проходить совсем иначе. Таким образом, не всегда модель с меньшим количеством переменных лучше.

Одним из преимуществ модели может быть наличие переменной, выбирая которую можно реализовать дихотомию при ветвлении по множеству решений. Поясним это на следующем примере.

Пример

Планируется открытие фабрики. Фабрика может размещаться на юге или на севере страны. Кроме того, нужно определить, будет ли конечный продукт упаковываться непосредственно на фабрике или нет.

Введем булевы переменные:

$$x_{sp} = \begin{cases} 1, & \text{если фабрика открывается на юге} \\ & \text{и упаковывает продукт,} \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$x_{sn} = \begin{cases} 1, & \text{если фабрика открывается на юге} \\ & \text{и не упаковывает продукт,} \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$x_{np} = \begin{cases} 1, & \text{если фабрика открывается на севере} \\ & \text{и упаковывает продукт,} \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$x_{nn} = \begin{cases} 1, & \text{если фабрика открывается на севере} \\ & \text{и не упаковывает продукт,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

Ограничение-равенство

$$x_{sp} + x_{sn} + x_{np} + x_{nn} = 1 \tag{4.41}$$

гарантирует, что будет реализован только один вариант работы фабрики.

Однако при таких переменных и ограничении процесс дихотомии при ветвлении, соответствующий тому, размещается ли фабрика на юге или на севере, на одной переменной организовать не удастся. Поэтому разумно ввести новую вспомогательную булеву переменную:

$$y = \begin{cases} 1, & \text{если фабрика открывается на севере,} \\ 0, & \text{если фабрика открывается на юге.} \end{cases}$$

Тогда вместо ограничения (4.41) будет пара ограничений:

$$x_{np} + x_{nn} - y = 0$$

и

$$x_{sp} + x_{sn} + y = 1.$$

Аналогично можно было ввести другую булеву переменную, чтобы ветвление соответствовало тому, будет ли продукт упаковываться или нет.

Пример [31]

Пусть в задаче имеется ограничение вида

$$\sum_j a_j x_j \leq b,$$

где все переменные x_j, коэффициенты a_j и правые части ограничений b — целые числа.

В процессе решения программными средствами это ограничение приводится к канонической форме. Для этого добавляется так называемая переменная недостатка u, $u \geq 0$, такая, что

$$\sum_j a_j x_j + u = b. \tag{4.42}$$

Однако это можно сделать вручную на этапе построения модели. Нетрудно заметить, что если все коэффициенты и переменные в ограничении целые числа, то u может принимать только целые значения. Наличие этой переменной позволяет организовать процесс ветвления по ней, а само ограничение (4.42) будет выступать в качестве отсекающей плоскости в соответствующей задаче LR [22].

4.4. Многогранники. Правильные неравенства

Рассмотрим следующую задачу IP:

$$\min fx$$

$$Ax \leq b,$$

$$x \geq 0, \text{ целые}.$$

Пусть X — множество допустимых решений этой задачи. Множество точек $P = \{x \in R^n | Ax \leq b\}$, удовлетворяющих конечному числу линейных неравенств, называют *многогранником.*

Многогранник P называют *представлением* множества допустимых решений X, если X совпадает с целочисленными точками из многогранника, т. е. $X = P \bigcap \mathbb{Z}^n$. На рис. 18 приводится два представления.

Пусть P^1 и P^2 — два представления для X. Как уже неоднократно наблюдалось в предыдущих разделах, для одной и той же задачи можно предложить

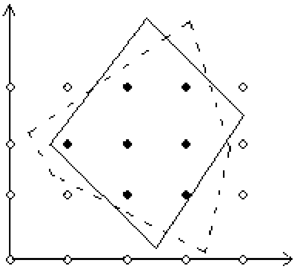

Рис. 18. Два представления

несколько математических моделей. Возникает вопрос: «Существует ли наилучшее представление и как его найти?»

Представление P^1 *лучше* P^2, если $P^1 \subset P^2$, поскольку допустимые целочисленные решения задачи лежат и в P^1, и в P^2, но с точки зрения линейной релаксации допустимая область представления P^2 больше, поэтому разрыв целочисленности с P^1 будет меньше, чем с P^2, и для задачи на минимум $z^*(X) \geq z^*(P^1) \geq z^*(P^2)$.

Выпуклой оболочкой множества X называется множество $conv(X)$, состоящее из точек вида $x = \sum_{i=1}^{T} \lambda_i x^i$, где $\sum_{i=1}^{T} \lambda_i = 1$, $\lambda_i \geq 0$, $i = 1, \ldots, T$, где $\{x^1, \ldots, x^T\}$ все точки из X, а T — общее число точек в множестве X. В этом случае точка x является выпуклой комбинацией точек $\{x^1, \ldots, x^T\}$. Выпуклая оболочка $conv(X)$ образует многогранник и является лучшим из возможных представлений для X (рис. 19). Суть выпуклой оболочки в том, что все крайние точки допустимого множества X содержатся в ней. Из теории линейного программирования известно, что оптимальное решение задачи LP достигается именно в крайних точках. Кроме того, $conv(X)$ содержится в любом множестве, содержащем X. Таким образом, $conv(X)$ — это минимальное множество, содержащее X и описываемое системой линейных неравенств, а значит модель задачи с представлением, в котором оно совпадает с выпуклой оболочкой, является самой лучшей моделью с точки зрения разрыва целочисленности. В этом случае разрыв равен нулю.

С практической точки зрения поиск такой переформулировки задачи сталкивается со следующими трудностями:

1) заранее неизвестно, как задать $conv(X)$, а ее нахождение является не менее сложной задачей, чем решить исходную задачу MIP;

2) для многих задач MIP описание $conv(X)$ требует экспоненциального чис-

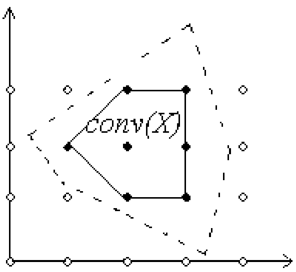

Рис. 19. Наилучшее представление

ла переменных и линейных неравенств, а значит для решения полученной таким образом задачи LP потребуется много времени. С другой стороны, добавление «правильных» ограничений улучшает верхнюю оценку LP и может заметно ускорить время работы алгоритма. Неравенства, добавление которых не меняет целочисленную допустимую область, называют *правильными неравенствами.*

Пример
Пусть допустимая область Y_1 задается следующими неравенствами:

$$y_1 \geq 1, \tag{4.43}$$

$$y_1 \leq 5, \tag{4.44}$$

$$y_1 + 0.8y_2 \leq 5.8, \tag{4.45}$$

$$y_1 - 0.8y_2 \geq 0.2, \tag{4.46}$$

$$y_1 + 8y_2 \leq 26, \tag{4.47}$$

$$y_1 \geq 0, \text{ целые}, \tag{4.48}$$

$$y_2 \geq 0, \text{ целые}. \tag{4.49}$$

На рис. 20 область Y_1 соответствует точкам с целочисленными координатами, лежащим внутри многогранника $ABCDEF$. В задаче линейной релаксации допустимая область представляет внутреннюю область многогранника $ABCDEF$. Рассмотрим область Y_2, которая формируется следующей системой неравенств:

$$y_1 + y_2 \leq 6, \tag{4.50}$$

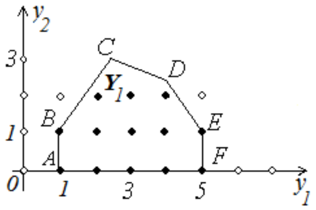

Рис. 20. Многогранник $ABCDEF$

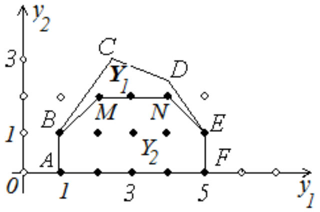

Рис. 21. Целочисленные допустимые решения

$$y_1 - y_2 \geq 0, \tag{4.51}$$

$$y_2 \leq 2, \tag{4.52}$$

$$y_1 \geq 0, \tag{4.53}$$

$$y_2 \geq 0. \tag{4.54}$$

Заметим, что множество целочисленных точек Y_1 и Y_2 совпадает (рис. 21) Следовательно, неравенства (4.50)–(4.52) — правильные.

Добавление правильных неравенств позволяет сузить допустимую область в задаче линейной релаксации, но не меняет допустимой области в задаче целочисленного программирования. На рис. 21 видно, что после добавления неравенств (4.50)–(4.52) к системе (4.43)–(4.49) допустимая область задачи линейной релаксации уменьшилась и совпадает с внутренней частью многогранника $ABMNEF$.

Пусть f — линейная целевая функция. На рис. 22 показано направление градиента f. Требуется найти $\max f$ при ограничениях (4.43)–(4.49). Мы уста-

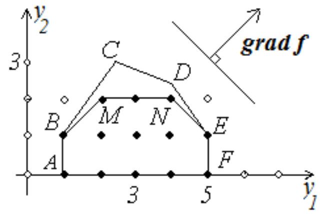

Рис. 22. Допустимая область и направление градиента

новили, что добавление неравенств (4.50)–(4.52) не меняет целочисленную область, следовательно, вместо исходной задачи можно решать задачу линейной релаксации (4.43)–(4.52). Оптимальное решение задачи линейной релаксации лежит в крайней точке N с целочисленными координатами. Таким образом, разрыв целочисленности равен нулю. Это означает, что оптимальное решение задачи MIP с целевой функцией f при ограничениях (4.43)–(4.49) совпадает с оптимальным решением задачи LP с целевой функцией f при ограничениях (4.43)–(4.52). Следовательно, в данном случае вместо задачи MIP можно решать задачу LP. Допустимая область, образованная (4.43)–(4.52), совпадает с $conv(Y_1 \bigcup Y_2)$.

В рассмотренном выше примере неравенство $y_1 + y_2 \leq 6$ — правильное, но оно не улучшает представление допустимого множества с точки зрения сокращения разрыва целочисленности. Неравенство $y_2 \leq 2$ не только является правильным, но еще и формирует $conv(Y_1)$. Возникает вопрос: «Как строить правильные неравенства, определяющие выпуклую оболочку?»

Построение правильных неравенств

Существует много способов построения правильных неравенств [25, 30], но универсального алгоритма не известно. Рассмотрим один из наиболее простых способов.

В общем случае под релаксацией множества X понимается любое надмножество Y, такое, что $X \subset Y$. Тогда любое правильное неравенство для множества Y является правильным неравенством для множества X. В качестве релаксации можно взять часть неравенств, определяющих область X и ограничения на значения переменных.

Пример

Вернемся к множеству Y_1, которое задается системой неравенств (4.43)–(4.49), и построим правильное неравенство. Мы уже заранее знаем ответ, это неравенство (4.50), которое обсуждалось выше.

Рассмотрим в качестве релаксации множества Y_1 множество Y, состоящее из точек, удовлетворяющих следующим неравенствам: (4.43)—(4.45), $0 \le y_2 \le 3$, (4.48) и (4.49). Неравенство $y_2 \le 3$ получается, если учесть, что $y_1 + 8y_2 \le 26$ и $y_1 \ge 0$, y_1 – целые, а следовательно, $y_2 \le \lfloor \frac{26}{8} \rfloor = 3$. Теперь, учитывая (4.45), заметим, что неравенство $y_1 + y_2 \le 6$ является правильным для Y, следовательно, оно будет правильным и для Y_1.

Большинство программных средств, прежде чем решать задачу, автоматически генерируют подобного рода правильные неравенства и добавляют их в исходную задачу. Заметим, что получение правильных неравенств не требует специальных знаний о структуре исходного множества допустимых решений.

Пример. Правильные неравенства для задачи планирования производства

Вернемся к задаче (4.25)–(4.30). Рассмотрим правильные неравенства для этой задачи.

Утверждение. Для любых l и C неравенства

$$\sum_{i \in C} y_i \le \sum_{i \in C} (\sum_{t=i}^{l} d_t) x_i + s_l, \ 1 \le l \le T, \ C \subseteq \{1, \dots, l\} \tag{4.55}$$

являются правильными для задачи (4.25)–(4.30).

Доказательство. Проверим, что добавление неравенств (4.55) в модель не приводит к потере допустимых решений задачи. Возьмем допустимое решение (y, s, x) задачи (4.26)–(4.30) и покажем, что неравенства (4.55) выполняются. Рассмотрим два случая.

1. $x_i = 0$ для любого $i \in C$.

Тогда $y_i = 0$ для любого $i \in C$ и из (4.55) получаем, что $s_l \ge 0$.

2. $x_i = 1$ для некоторого $i \in C$.

Пусть $k = \min\{i \in C | x_i = 1\}$. Тогда $y_i = 0$ для любого $i < k$ и $\sum_{i \in C} y_i \le \sum_{t=k}^{l} y_t = \sum_{t=k}^{l} d_t + s_l - s_{k-1} \le \sum_{t=k}^{l} d_t + s_l \le \sum_{i \in C} (\sum_{t=i}^{l} d_t) x_i + s_l$. Что и требовалось доказать.

Известен и более сильный результат.

Теорема [30]. *Выпуклая оболочка множества* (4.26)–(4.30) *задается следующей системой ограничений:*

$$y_1 = d_1 + s_1, \tag{4.56}$$

$$s_{t-1} + y_t = d_t + s_t, \quad t = 2, \dots, T, \tag{4.57}$$

$$s_T = 0, \tag{4.58}$$

$$x_t \le 1, \quad t = 1, \dots, T, \tag{4.59}$$

$$\sum_{i \in C} y_i \le \sum_{i \in C} (\sum_{t=i}^{l} d_t) x_i + s_l \ \textit{для любого } l, \text{ и } C \ne \emptyset, \tag{4.60}$$

$$y_t \ge 0, s_t \ge 0, x_t \ge 0, \quad t = 1, \dots, T. \tag{4.61}$$

4.5. Целочисленные решения задачи линейного программирования

Структура некоторых задач, например, задачи о назначениях, задачи о нахождении потока минимальной стоимости и других, такова, что оптимальное решение соответствующей задачи линейного программирования является целочисленным. Это важная характеристика задачи. Если все решения задачи линейного программирования — целочисленные, то обычный аппарат линейного программирования позволяет найти оптимальное решение, удовлетворяющее требованию целочисленности и являющееся, следовательно, также оптимальным решением соответствующей задачи целочисленного линейного программирования. Причина такой благоприятной ситуации может быть в том, что система ограничений обладает определенными свойствами, о которых сейчас пойдет речь.

Квадратная матрица $D = (d_{ij})$ с целочисленными компонентами, $i = 1, \ldots, m$, $j = 1, \ldots, m$ называется *унимодулярной*, если ее определить равен ± 1.

Матрица $A = (a_{ij})$ с целочисленными компонентами, $i = 1, \ldots, m$, $j = 1, \ldots, n$ называется *полностью унимодулярной*, если определитель любой ее квадратной подматрицы равен 0 или ± 1.

Таким образом, полностью унимодулярная матрица состоит из 0, 1 или -1. Однако, например, матрица

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

не является полностью унимодулярной, поскольку определитель подматрицы, составленной из первых трех столбцов и первых трех строк, равен —2.

К сожалению, определение не дает эффективного способа проверки, является ли матрица полностью унимодулярной, так как вычисление определителей всех квадратных подматриц матрицы A — трудоемкая процедура. Следующая теорема, которая является непосредственным следствием из определения, описывает варианты построения других унимодулярных матриц из заданной.

Теорема [30]. *Следующие утверждения эквиваленты.*

1. *Матрица A является полностью унимодулярной.*

2. *Матрица, полученная транспонированием матрицы A, является полностью унимодулярной.*

3. *Матрица, полученная приписыванием к A единичной матрицы E справа, является полностью унимодулярной.*

4. *Матрица, полученная удалением строки (столбца) из матрицы A, является полностью унимодулярной.*

5. *Матрица, полученная умножением строки (столбца) матрицы A на (-1), является полностью унимодулярной.*

6. *Матрица, полученная перестановкой двух строк (столбцов) матрицы A, является полностью унимодулярной.*

7. *Матрица, полученная дублированием строк (столбцов) матрицы A, является полностью унимодулярной.*

8. *Матрица, полученная после выполнения операции замены базисной переменной, является полностью унимодулярной.*

Следующее достаточное условие гарантирует, что матрица A будет полностью унимодулярной.

Утверждение [13]. *Матрица A, составленная из 0, 1 или −1, является полностью унимодулярной, если в каждом столбце не более двух ненулевых элементов, и строки матрицы A можно так разбить на два подмножества I_1 и I_2, что для каждого столбца верно следующее:*

1. *Если два ненулевых элемента имеют одинаковые знаки, то соответствующие им строки лежат в разных множествах.*

2. *Если два ненулевых элемента имеют разные знаки, то соответствующие им строки лежат в одном и том же множестве.*

Рассмотрим задачу LP в следующем виде:

$$\max \sum_{j=1}^{n} c_j x_j \tag{4.62}$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1, \ldots, m \tag{4.63}$$

$$x_j \geq 0, j = 1, \ldots, n, m < n. \tag{4.64}$$

Каким условиям должна удовлетворять матрица $A = (a_{ij})$ и вектор правых частей $b = (b_1, \ldots, b_m)^T$, чтобы все вершины многогранника (4.63)–(4.64) были целочисленными? Ответ на этот вопрос дает следующая теорема.

Теорема [30]. *Пусть целочисленная матрица $A = (a_{ij})$, $i = 1, \ldots, m$, $j = 1, \ldots, n, m < n$ имеет ранг m и вектор правых частей $b = (b_1, \ldots, b_m)^T$ — целочисленный. Для того, чтобы вершины многогранника (4.63)–(4.64) лежали в целочисленных точках, необходимо и достаточно, чтобы матрица A была полностью унимодулярной.*

Пример. Транспортная задача

Пусть имеется m пунктов производства и n пунктов потребления однородного продукта. Объем производства в пункте i равен a_i; объем потребления в пункте j равен b_j, $i = 1, \ldots, m$, $j = 1, \ldots, n$. Предполагается, что производство и потребление сбалансированы, т. е.

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j. \tag{4.65}$$

Задана матрица $c = (c_{ij})$ транспортных расходов на перевозку единицы продукции из пункта производства i в пункт потребления j. Требуется составить план перевозок, который, не нарушая производственные мощности, удовлетворяет все потребности с минимальными суммарными затратами на перевозки.

Введем переменные $x_{ij} \geq 0$, обозначающие объем перевозок из пункта производства i в пункт потребления j. Тогда получаем следующую задачу линейного программирования:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \tag{4.66}$$

при ограничениях:

$$\sum_{j=1}^{n} x_{ij} = a_i, i = 1, \ldots, m, \tag{4.67}$$

$$\sum_{i=1}^{m} x_{ij} = b_j, j = 1, \ldots, n, \tag{4.68}$$

$$x_{ij} \geq 0, i = 1, \ldots, m, j = 1, \ldots, n. \tag{4.69}$$

Условие сбалансированности производства и потребления является необходимым и достаточным условием разрешимости транспортной задачи [13]. Число переменных x_{ij} в этой задаче равно mn, число ограничений равно $m + n$. Из-за условия сбалансированности производства и потребления строки (4.67) и (4.68) являются линейно зависимыми, следовательно, ранг системы ограничений равен $(m + n - 1)$, а значит число ненулевых значений x_{ij} не превосходит $(m + n)$.

Если объемы производства a_i и объемы потребления b_j — целые числа, то любая вершина многогранника (4.67)–(4.69) целочисленная, поскольку матрица ограничений является полностью унимодулярной [4].

Пример

Пусть в модели среди всех ограничений имеется одно, которое нарушает унимодулярность всей матрицы. Рассмотрим вариант модификации этого ограничения в модели, который приводит к полностью унимодулярной подматрице.

Пусть в модели имеются булевы переменные x_i, $i = 1, \ldots, n$ и задача требует, чтобы выполнялось следующее условие:

$$\text{если } x_n = 0, \text{ то } x_1 = 0, x_2 = 0, \ldots, x_{n-1} = 0. \tag{4.70}$$

Тогда неравенство

$$x_1 + x_2 + \ldots + x_{n-1} \le (n-1)x_n \tag{4.71}$$

гарантирует выполнение этого условия. Нетрудно заметить, что условие (4.70) можно смоделировать и по-другому, системой из $(n-1)$ неравенств:

$$\begin{aligned} x_1 &\le x_n, \\ x_2 &\le x_n, \\ &\vdots \\ x_{n-1} &\le x_n. \end{aligned} \tag{4.72}$$

Построив для модели с представлением (4.72) двойственную задачу [14], нетрудно заметить, что матрица ограничений соответствующей двойственной задачи будет полностью унимодулярной. Следовательно, решив линейную релаксацию двойственной задачи, можно найти оптимальное целочисленное решение исходной задачи, пользуясь средствами линейного программирования. Таким образом, замена одного ограничения (4.71) системой ограничений (4.72) улучшает модель.

4.6. Уточнение значения границ переменных

Рассмотрим распространенное в моделях MIP ограничение вида $y \le ux$, где u — некоторое положительное число, $x \in \{0, 1\}$. Насколько важно значение u с точки зрения разрыва целочисленности?

Если переменная x принимает целые значения, то неважно, чему равно u, но если $0 \le x \le 1$, то значение u оказывается существенным.

Предположим, что максимальное допустимое значение, которое может принимать переменная y, равно u' и $u' < u$, а в целевой функции имеются слагаемые вида fx, соответствующие, например, затратам на запуск производства, $f > 0$. Если в оптимальном решении $y = u'$, то, учитывая ограничение $y \le ux$, получаем, что $x = \frac{u'}{u}$ и $x < 1$. Следовательно, затраты составят $f\frac{u'}{u}$. С другой стороны, если бы ограничение было записано как $y \le u'x$, то при $y = u'$ получили бы $x = 1$ и затраты составили бы величину f. Следовательно, чем грубее значение u по сравнению с u', тем больше разрыв целочисленности.

Иногда хорошие граничные условия можно определить аналитически.

Пример

В задаче о планировании производства, рассмотренной в разд. 4.2, в ограничениях (4.28) для каждого промежутка времени t вместо грубой оценки $\sum_{k=1}^{T} d_k$ лучше использовать уточненную оценку $\sum_{k=t}^{T} d_k$.

Пример. Задача о потоке с фиксированными затратами

Рассмотрим на примере задачи о потоке с фиксированными затратами аналитические способы уточнения границ значений переменных. Пусть имеется сеть, как показано на рис. 23.

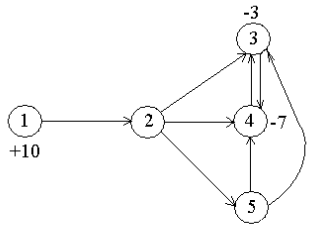

Рис. 23. Сеть для задачи о потоке с фиксированными затратами

Известны c_{ij} — удельные стоимости перевозки продукции вдоль дуги (i, j), h_{ij} — плата за использование дуги (i, j), b_i — спрос или предложение в вершине i. Требуется доставить 3 и 7 единиц продукции в 3-ю и 4-ю вершины соответственно с минимальными затратами.

Введем неотрицательные переменные x_{ij} — величина потока, пропускаемого по дуге (i, j);

$$y_{ij} = \begin{cases} 1, & \text{если по дуге } (i,j) \text{ выполняются перевозки,} \\ 0 - & \text{в противном случае.} \end{cases}$$

Пусть B — некоторое положительное большое число.
Математическая модель выглядит следующим образом:

$$\min \sum_{(i,j)\in A} (c_{ij}x_{ij} + h_{ij}y_{ij}), \tag{4.73}$$

при ограничениях:

$$x_{ij} \le By_{ij}, (i,j) \in A, \tag{4.74}$$

$$\sum_{j\in V} x_{ji} - \sum_{j\in V} x_{ij} = b_i, i \in V, \tag{4.75}$$

$$x_{ij} \ge 0, (i,j) \in A, \tag{4.76}$$

$$y_{ij} \in \{0,1\}, (i,j) \in A. \tag{4.77}$$

В данной задаче отсутствуют ограничения на пропускные способности дуг, поэтому в ограничениях (4.74) присутствует некоторое большое число B, что является недостатком данной модели. Уточним верхние оценки на принимаемые значения для переменных x_{ij}, проанализировав веса вершин сети.

Итак, вес первой вершины составляет 10 единиц. Если дуга $(1, 2)$ используется, то по дуге x_{12} можно отправить не более 10 единиц, следовательно,

$$x_{12} \le 10y_{12}.$$

Вторая вершина — транзитная, и доставить в нее можно не более 10 единиц, следовательно, и вывезти из нее по каждой выходящей дуге можно не более 10 единиц, значит

$$x_{23} \leq 10y_{23},$$

$$x_{24} \leq 10y_{24},$$

$$x_{25} \leq 10y_{25}.$$

В третьей вершине потребляется 3 единицы продукции, поэтому если доставить максимально возможное количество продукции в размере 10 единиц в третью вершину, то вывезти из нее можно будет не более 7 единиц:

$$x_{34} \leq 7y_{34}.$$

Аналогично получаем следующие уточненные границы на значения переменных:

$$x_{43} \leq 3y_{43},$$

$$x_{54} \leq 10y_{54},$$

$$x_{53} \leq 10y_{53}.$$

Добавление новых неравенств вместо ограничений (4.74) позволяет сузить допустимую область и сократить разрыв.

4.7. Удаление избыточных ограничений

Рассмотрим еще один способ предварительной обработки математической модели с булевыми переменными. Пусть допустимое множество задается группой линейных неравенств вида

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

$$x_j \in \{0, 1\}, j = 1, \dots, n.$$

Если $a_j < 0$ для некоторого j, тогда заменой переменной x_j на $(1 - x'_j)$ можно получить эквивалентное ограничение:

$$\sum_{j=1,\dots,n|a_j>0} a_j x_j + \sum_{j=1,\dots,n|a_j<0} |a_j| x'_j \leq b - \sum_{j=1,\dots,n|a_j<0} a_j.$$

Это сведение позволяет без ограничения общности считать, что все $a_j > 0$, $j = 1, \dots, n$. Если $\sum_{j \in N} a_j > b$, для некоторого $N \subseteq \{1, \dots, n\}$, то в модель можно ввести дополнительное неравенство

$$\sum_{j \in N} x_j \leq |N| - 1 \tag{4.78}$$

или несколько неравенств для каждого $k \in N$, при котором $\sum_{j \in N \setminus \{k\}} a_j \le b$.

Возможно, что после таких преобразований удастся объединить некоторые неравенства и зафиксировать значения некоторых переменных, сократить размерность задачи.

Пример

Рассмотрим следующую систему неравенств:

$$-3x_2 - 2x_3 \le -2, \tag{4.79}$$

$$-4x_1 - 3x_2 - 3x_3 \le -6, \tag{4.80}$$

$$2x_1 - 2x_2 + 6x_3 \le 5, \tag{4.81}$$

$$x_i \in \{0, 1\}, i = 1, 2, 3. \tag{4.82}$$

Выполним замену переменных, получим:

$$3x_2' + 2x_3' \le 3, \tag{4.83}$$

$$4x_1' + 3x_2' + 3x_3' \le 4, \tag{4.84}$$

$$2x_1 + 2x_2' + 6x_3 \le 7, \tag{4.85}$$

$$x_1, x_i' \in \{0, 1\}, i = 1, 2, 3. \tag{4.86}$$

Учитывая (4.78) для (4.83), получаем

$$x_2' + x_3' \le 1 \text{ или } x_2 + x_3 \ge 1, \tag{4.87}$$

для (4.85) получаем

$$x_2' + x_3 \le 1 \text{ или } x_3 \le x_2. \tag{4.88}$$

Объединяя (4.87) и (4.88), получаем $x_2 = 1$ или $x_2' = 0$, и тогда ограничение (4.79) избыточное, а ограничения (4.84) и (4.85) превращаются в

$$4x_1' + 3x_3' \le 4 \text{ и } 2x_1 + 6x_3 \le 7.$$

Учитывая (4.78), получаем

$$x_1' + x_3' \le 1 \text{ или } x_1 + x_3 \ge 1$$

и

$$x_1 + x_3 \le 1,$$

следовательно, $x_1 + x_3 = 1$. Используя это равенство, можно исключить одну из переменных x_1 или x_3. Таким образом, удается сократить число переменных и ограничений.

На этом изложение идей и фактов о том, как строить модели, оценивать их качество и выбирать среди возможных моделей наилучшую, завершается. Заинтересованный читатель может детально познакомиться с предложенными техниками из литературы, приведенной в конце пособия. В следующем разделе предлагаются упражнения, направленные на усвоение прочитанного материала.

5. Упражнения

В этом разделе собраны упражнения на моделирование, поиск наилучшего представления допустимого множества и построение правильных неравенств. Упражнения разделены на две части. В первой части собраны задачи, которые решаются без помощи специальных программных средств. Во второй части приводятся задания, для выполнения которых сначала рекомендуется построить математическую модель, а затем найти решение с помощью программных средств. Существует много коммерческих и бесплатных программных обеспечений для решения оптимизационных задач [16 — 20]. Читатель может выбрать любой из них, например, систему GAMS. Размерность задач в упражнениях подобрана таким образом, что можно построить математическую модель, количество ограничений и целочисленных переменных которой будет не слишком велико. Для тех, кто незнаком с этой системой, в следующем разделе разобран пример решения задачи в GAMS.

5.1. Теоретические задания

1. Пусть $x_i \in \{0,1\}, i \in \{1,\dots,n\}$. Что можно сказать о значениях каждой из переменных, удовлетворяющих следующим неравенствам:

   1) $x_i + x_j \le 1$ и $x_i \le x_j$, для некоторых $i,j \in \{1,\dots,n\}$,
   2) $x_i + x_j \le 1$ и $x_i + x_j \ge 1$, для некоторых $i,j \in \{1,\dots,n\}$,
   3) $x_i \le x_j$ и $x_j + x_k \le 1$, для некоторых $i,j,k \in \{1,\dots,n\}$?

2. Пусть область S задается следующим образом:

$$\sum_{j\in N_1} a_j x_j - \sum_{j\in N_2} a_j x_j \le b,$$

$$x_j \in \{0,1\}, j \in \{N_1\} \cup \{N_2\},$$

   где $a_j > 0$. Выпишите необходимые и достаточные условия, чтобы

   1) $S = \emptyset$,
   2) S совпадало со всеми 0–1 векторами размерности $|N_1| + |N_2|$,
   3) $x_j = 0$ для всех $x \in S$,
   4) $x_j = 1$ для всех $x \in S$,
   5) $x_i + x_j \le 1$ для всех $x \in S$,
   6) $x_i \le x_j$ для всех $x \in S$,
   7) $x_i + x_j \ge 1$ для всех $x \in S$.

3. Представьте произведение $y_1 y_2 y_3$ трех (четырех и т.д.) булевых переменных линейными ограничениями.

4. Пусть $x_j \geq 0, \forall j$. Задайте выполнение по крайней мере одного из двух неравенств
$$\sum_j a_{1j}x_j \leq b_1 \text{ или } \sum_j a_{2j}x_j \leq b_2,$$
используя только линейные ограничения и одну вспомогательную булеву переменную.

5. Пусть $x_j \geq 0, \forall j$. Задайте выполнение одного из двух неравенств
$$\sum_j a_{1j}x_j \leq b_1 \text{ или } \sum_j a_{2j}x_j \leq b_2,$$
используя только линейные ограничения и одну вспомогательную булеву переменную.

6. Задайте выполнение одного из n неравенств
$$\sum_j a_{1j}x_j \leq b_1$$
или
$$\sum_j a_{2j}x_j \leq b_2,$$
или
$$\vdots$$
или
$$\sum_j a_{nj}x_j \leq b_n,$$
используя только линейные ограничения и n вспомогательных булевых переменных.

7. Пусть кусочно-линейная функция $f(x)$ задана следующим образом:
$$f(x) = \begin{cases} 0, & \text{если } x = a_0, \\ c_0(x - a_0), & \text{если } a_0 < x \leq a_1, \\ c_0a_1 + c_1(x - a_1), & \text{если } a_1 < x \leq a_2, \\ c_0a_1 + c_1(a_2 - a_1) + c_2(x - a_2), & \text{если } a_2 < x \leq a_3, \end{cases}$$
где $c_0 = 50$, $c_1 = 207$, $a_0 = 0$, $a_1 = 140$, $a_2 = 220$, $a_3 = 420$. Постройте MIP поиска минимума $f(x)$, $x \in [a_0, a_3]$.

8. Смоделируйте с помощью линейный ограничений, чтобы переменная y принимала значение, равное $\max(u_1, u_2)$, где $u_1, u_2 \geq 0$.

9. Совет директоров должен принять решение о выборе инвестиций. Имеется 7 вариантов вложений. На множестве инвестиций заданы условия вложений (табл. 6).

Таблица 6

Условия инвестиций

Тип инвестиции	Условия вложений
1	—
2	если инвестировали в 1
3	если инвестировали в 2
4	если инвестировали и в 1, и в 2
5	если не было инвестиций в 1 или в 2
6	если не было инвестиций ни в 2, ни в 3
7	если были инвестиции в 2, и не было в 3

Известны r_i — доход от инвестиции i, c_i — расходы при вложении i. Объем инвестиций не должен превышать M у.е. Компания стремится получить максимальную прибыль от вложений. Постройте математическую модель.

10. Компания должна назначить 8 команд на 3 проекта: A, B и C. Доход от каждого проекта зависит от числа команд, назначенных на проект. Ни один проект не должен остаться без команды. Выполнять один проект может не более 5 команд. Каждая команда может быть назначена только на один проект. Доход от соответствующих назначений приводится в табл. 7.

Таблица 7

Доход

Количество команд	Проект A	Проект B	Проект C
одна	45	20	50
две	70	45	70
три	90	75	80
четыре	105	110	100
пять	120	150	130

Цель компании — назначить команды на проекты так, чтобы максимизировать суммарный доход.

Ответьте на следующие вопросы.
а. Сколько ограничений нужно включить в модель:

1) одно;
2) два или три;

3) четыре или пять;

4) по крайней мере шесть?

б. Какое из следующих утверждений неверно:

1) все коэффициенты при переменных во всех ограничениях либо 0, либо 1, либо -1;

2) в модели все переменные — булевы;

3) можно записать модель, используя ограничения только в виде равенств;

4) в оптимальном решении на проекты назначены все 8 команд?

в. Предположим, что появляется еще один проект. Сколько потребуется новых переменных:

1) 0;

2) 1;

3) 3;

4) 5?

г. Вернитесь к исходной задаче. Предположим, что появляется еще одна команда. Сколько потребуется новых переменных:

1) 0;

2) 1;

3) 3;

4) 5?

11. Для обслуживания m клиентов могут быть задействованы n грузовиков. Каждого клиента можно посетить только один раз. Время на дорогу от клиента i до клиента j, которое затрачивает грузовик k, равно c_{ijk}, $i, j \in \{1, \ldots, m\}$, $k \in \{1, \ldots, n\}$. Общее время, в течение которого может быть задействован грузовик k, не должно превышать величину l_k. Постройте модель целочисленного линейного программирования допустимой схемы объезда всех клиентов.

12. Для обслуживания склада используется два грузовика A и B. Каждый грузовик может выезжать в 13:00, 14:00, 15:00 или в 16:00. Грузовик B не должен выезжать в течение часа после выезда грузовика A. Например, если грузовик A выехал в 13:00, то грузовик B может выезжать в 15:00 или в 16:00. Каждый грузовик выезжает со склада в течение дня один раз.

Используя переменные

$$x_i = \begin{cases} 1, & \text{если грузовик } A \text{ выехал в момент времени } i, \\ 0 \ - & \text{в противном случае,} \end{cases}$$

$$y_i = \begin{cases} 1, & \text{если грузовик } B \text{ выехал в момент времени } i, \\ 0 \ - & \text{в противном случае,} \end{cases}$$

предложите два разных способа построения множества допустимых выездов грузовиков и сравните их.

13. Компания занимается доставкой посылок. Нужно развести посылки 10 клиентам. Заданы величины d_j — количество посылок, которые нужно доставить клиенту j, $j = 1, \ldots, 10$. В компании имеется 4 грузовика вместимостью q_i посылок каждый, $i = 1, \ldots, 4$. Предполагается, что все посылки не входят в один грузовик, но $\sum_{i=1}^{4} q_i \geq \sum_{j=1}^{10} d_j$ и $d_j \leq q_i$. Партию посылок предназначенную одному клиенту нельзя доставлять разными грузовиками. Затраты при отправке грузовика i равны c_i, $i = 1, \ldots, 4$. Каждый грузовик может объехать не более пяти клиентов, кроме того, к следующим парам клиентов нельзя доставлять посылки одним и тем же грузовиком: (1,7), (2,6), (2,9). Определите, какие грузовики необходимо задействовать, чтобы доставить все посылки клиентам с минимальными суммарными затратами. Постройте математическую модель.

14. Отдел статистики подсчитал, что в городе имеется n женихов и n невест. Известна привлекательность (c_{ij}) невесты i женихом j. Постройте математическую модель так, чтобы каждому жениху назначить невесту и каждой невесте назначить жениха с максимальной суммарной привлекательностью между парами. Правда ли, что задача является полиномиально разрешимой?

15. Компания производит два типа продукции на одном заводе, чтобы удовлетворить запросы клиентов в пяти регионах. Поставка продукции осуществляется через два крупных центра. Величина d_{jk} определяет запрос клиентов из региона j на продукт k, $k = 1, 2$, $j = 1, \ldots, 5$. Компания должна решить, какой тип продукта в какой центр доставить и как его распределить по регионам, чтобы минимизировать суммарные затраты. Затраты складываются из:

   - фиксированных затрат f_{ik}, возникающих, если продукт типа k поставляется из центра i;
   - фиксированных затрат g_{ijk}, возникающих, если продукт типа k поставляется доставляется клиенту в регион j через центр i;
   - удельных затрат c_{ijk} на транспортировку продукта типа k в регион j через центр i.

Постройте математическую модель.

16. У авиакомпании имеется расписание вылета воздушных судов в пять городов на ближайший день. Для обслуживания самолетов необходимо нанять экипажи и составить для них расписание. Время работы каждого экипажа в течение суток ограничено. Необходимо определить, какое минимальное количество экипажей нужно нанять и составить для них расписание, чтобы все запланированные вылеты состоялись. Постройте математическую модель, используя задачу о покрытии.

17. Имеется n работ и одна машина для выполнения этих работ. В каждый момент времени может выполняться только одна работа. Каждая работа выполняется без прерывания. Известны следующие целочисленные величины:
p_j — длительность выполнения работы j,
d_j — крайний срок завершения каждой работы и w_j — важность работы, $j = 1, \ldots, n$.

    1) Требуется найти допустимое расписание, минимизирующее взвешенную сумму времен завершения работ. Постройте математическую модель. Избегайте в модели введения вспомогательных больших чисел, используйте точные оценки на время окончания работ.

    2) Постройте модель, используя следующие переменные:

$$x_{jt} = \begin{cases} 1, & \text{если работа } j \text{ завершилась в момент} \\ & \text{времени } t, \\ 0 \ - & \text{в противном случае.} \end{cases}$$

18. Компания имеет две машины для производства пластиковых бутылок. В течение года нужно проводить техническое обслуживание оборудования. Техническое обслуживание каждой машины длится 2 месяца подряд. Кроме того, в декабре и январе половина рабочих отдыхает, поэтому в производственном процессе может быть задействована только одна машина, а другая машина в это время может проходить техническое обслуживание. Ежемесячный спрос на бутылки составляет d_t штук, $t = 1, \ldots, 12$. Максимальное количество бутылок, которые машина k может произвести в месяц составляет a_k штук, $k = 1, 2$. Для того, чтобы произвести a_k бутылок, машина должна проработать l_k рабочих дней. В каждом месяце только L_t рабочих дней, $t = 1, \ldots, 12$.

    Необходимо составить допустимый график проведения технического обслуживания машин при условии, что спрос на бутылки будет удовлетворен. Постройте математическую модель.

    Как изменится модель, если требуется:
    1) минимизировать суммарные ежемесячные скачки в загруженности рабочих дней;

2) минимизировать наибольший скачок в течение года между загруженными рабочими днями.

19. Завод производит N разных продуктов из злаков и фруктов. Производственный процесс состоит из трех этапов: подготовка сырья (очистка, сушка и т. д.), смешивание (расфасовка) и упаковка. На рис. 24 изображена общая схема работы завода. Сырье поставляется в неограниченном

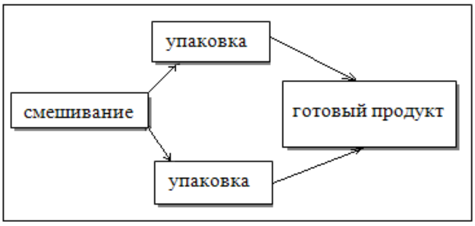

Рис. 24. Схема работы завода

количестве. Для смешивания и расфасовки фруктов и злаков используется одна машина. После этого готовый продукт упаковывается на одной из двух машин.

Время работы каждой машины k ограничено величиной L^k, производительность машины k равна α^{ik} единиц продукта i в час, $i = 1, \ldots, N$, $k = 1, 2, 3$.

После смешивания и расфасовки продукта i машину необходимо промыть в течение β^i часов. Завод имеет склад, и первоначально на складе имеется S_0^i единиц упакованного продукта i, а к концу каждой недели t на складе должно оставаться S_t^i единиц упакованного продукта i в качестве резерва, $i = 1, \ldots, N$.

Необходимо спланировать производство продуктов на T недель вперед так, чтобы на складе оставалось как можно меньше готовой продукции за весь период планирования, при этом удовлетворить спрос D_t^i единиц упакованного продукта i на неделю t. Постройте математическую модель.

20. Вспомните два представления допустимого множества решений задачи коммивояжера (разд. 3.2). Какое из этих представлений лучше?

21. Вспомните задачу коммивояжера, предложенную в разд. 3.2, и предло-

жите представление, используя следующие переменные:

$$x_{ijk} = \begin{cases} 1, & \text{если дуга } (i,j) \text{ идет } k\text{-й по порядку в обходе,} \\ 0 \ - & \text{в противном случае.} \end{cases}$$

22. Вспомните двухстадийную задачу о гильотинном раскрое, рассмотренную в разд. 3. Постройте математическую модель при условии, что прямоугольники можно поворачивать на 90^0.

23. Вспомните задачу о башнях, рассмотренную в разд. 3. В том примере пять из шести возможных вариантов размещения башен приводят к разбросу клиентов, равному двум, и только один вариант с открытием 2-й и 3-й башен дает равномерное распределение клиентов с перепадом, равным нулю. Придумайте пример, т. е. подберите соответствующие значения n, m и p, и задайте матрицу (c_{ij}), демонстрирующую, что перепад между максимальным и минимальным количеством клиентов может быть большим.

24. Задан взвешенный неориентированный граф $G = (V, E)$ со множеством вершин V и множеством ребер E. Каждому ребру $e \in E$ приписан неотрицательный вес w_e. Под весом разреза для произвольного разбиения множества вершин на непересекающиеся подмножества будем понимать сумму весов ребер, каждое из которых имеет вершины в разных подмножествах. Для заданного натурального числа p задача заключается в том, чтобы найти разбиение множества вершин V на подмножества мощности не более p, имеющее минимальный вес разреза.

    Представьте задачу в терминах целочисленного линейного программирования. Имеет ли полученная математическая модель симметрии? Если да, то удалите симметрии введением дополнительных ограничений.

25. Любитель фильмов обнаружил в компьютерной сети новую библиотеку с фильмами и решает просмотреть все фильмы. Он знает, сколько нужно времени для того, чтобы скачать, просмотреть один фильм, и размер каждого фильма. Размер доступного пространства для хранения видеофайлов на диске его компьютера ограничен. Поэтому любитель сначала полностью скачивает фильм, затем просматривает его и удаляет с диска. Скачивание и просмотр видео происходит без прерываний. Пока один фильм просматривается, другой фильм может закачиваться на диск. Скачивание очередного фильма начинается, только если перед началом скачивания он может целиком поместиться на диск.

    Требуется найти порядок скачивания и просмотра фильмов, чтобы закончить просмотр всех фильмов как можно быстрее.

    Постройте математическую модель целочисленного программирования. Правда ли, что среди оптимальных решений задачи всегда найдется решение с одинаковым порядком скачивания и просмотра фильмов?

26. Заданы матрица (h_{ij}), $i = 1, \ldots, n$, $j = 1, \ldots, m$ с положительными вещественными элементами и вектор (c_j), $j = 1, \ldots, m$ с положительными вещественными элементами. Пусть $\mathfrak{F}$ — семейство всех непустых подмножеств множества $\{1, \ldots, m\}$. Для каждого $F \in \mathfrak{F}$ определим

$$z(F) = \sum_{i=1}^{n} \max_{j \in F} h_{ij} - \sum_{j \in F} c_j.$$

Постройте для задачи максимизации функции $z(F)$ модель целочисленного линейного программирования, используя только булевы переменные.

27. Докажите, что если все компоненты матрицы A и вектор-столбца b целочисленные и A — унимодулярная матрица, то все базисные решения задачи

$$\min cx$$
$$Ax = b$$
$$x \geq 0$$

имеют целочисленные компоненты. Докажите, что утверждение верно, если $Ax \leq b$ $(Ax \geq b)$.

28. Множество S может быть задано несколькими способами:

$$97x_1 + 32x_2 + 25x_3 + 20x_4 \leq 139,$$
$$x_i \in \{0, 1\}, i = 1, \ldots, 4$$

или

$$2x_1 + x_2 + x_3 + x_4 \leq 3,$$
$$x_i \in \{0, 1\}, i = 1, \ldots, 4$$

или

$$x_1 + x_2 + x_3 \leq 2,$$
$$x_1 + x_3 + x_4 \leq 2,$$
$$x_1 + x_2 + x_4 \leq 2,$$
$$x_i \in \{0, 1\}, i = 1, \ldots, 4.$$

Какой из способов наиболее эффективный при решении задачи $\max_{x \in S} cx$?

29. Пользуясь результатами из упражнений 1 и 2, решите следующую задачу, не делая перебора:

$$\max 2x_1 - 2x_2 + 3x_3 + x_4$$

при условиях:

$$7x_1 + 3x_2 + 9x_3 - 2x_4 + 2x_5 \leq 7,$$
$$-6x_1 + 2x_2 - 3x_3 + 4x_4 + 9x_5 \leq -2,$$
$$x_j \in \{0, 1\}, j = 1, \ldots, 5.$$

30. Пусть множество S задается следующей системой неравенств:

$$x_1 - x_2 \leq 1,$$

$$4x_1 + x_2 \leq 28,$$

$$x_1 + 4x_2 \leq 27,$$

$$x_1 \geq 0, x_2 \geq 0, \text{ целые.}$$

Найдите систему неравенств, задающую выпуклую оболочку множества S.

31. Многогранник P задается следующей системой неравенств:

$$x_1 - x_2 \leq 0,$$

$$-x_1 + x_2 \leq 1,$$

$$2x_2 \geq 5,$$

$$8x_1 - x_2 \leq 16,$$

$$x_1 + x_2 \geq 4,$$

$$x_1 \geq 0, x_2 \geq 0, \text{ целые.}$$

Найдите «наименьшее» представление для многогранника P.

32. Найдите $conv(X)$, если множество X задано следующим образом:

$$-x_1 + 2x_2 \leq 7,$$

$$x_1 + 3x_2 \leq 15,$$

$$7x_1 - 3x_2 \leq 23,$$

$$x_1 \geq 0, x_2 \geq 0, \text{ целые.}$$

33. Найдите наилучшее представление для множества X, заданного следующим образом:

$$-x_1 + x_2 \leq 3,$$

$$x_2 \leq 4,$$

$$7x_1 + 11x_2 \leq 65,$$

$$7x_1 - 3x_2 \leq 23,$$

$$x_1 \geq 0, x_2 \geq 0, \text{ целые.}$$

34. Пусть многогранник задается следующей системой неравенств:

$$x_1 + x_2 \le b_1,$$

$$-x_1 + x_2 \le b_2,$$

$$x_1 \ge 0, x_2 \ge 0,$$

где $b = (b_1, b_2)^T$ — целочисленный вектор с неотрицательными компонентами.
Ответьте на следующие вопросы.
1. Является ли матрица ограничений полностью унимодулярной?
2. Существует ли вектор правых частей b, такой что все вершины многогранника будут целочисленными точками?

35. Рассмотрим допустимую область задачи, известной в литературе как задача размещения с ограничениями на мощности производства:

$$\sum_{j \in J} y_{ij} = a_i \text{ для каждого } i \in I, \tag{5.1}$$

$$\sum_{i \in I} y_{ij} \le b_j x_j \text{ для каждого} j \in J, \tag{5.2}$$

$$y_{ij} \ge 0, x_j \in \{0, 1\} \text{ для каждого } i \in I, j \in J. \tag{5.3}$$

Пусть $I' \subseteq I$ и $z_j = \sum_{i \in I'} y_{ij}$, такие, что

$$\sum_{j \in J} z_j = \sum_{i \in I'} a_i \text{ и } z_j \le b_j x_j.$$

Постройте правильные неравенства для (5.1)–(5.3).

36. Рассмотрим множество:

$$X = \{y \in \mathbb{Z}_+^n : C \sum_{j=1}^{n} y_j \ge b\}, \tag{5.4}$$

где C, b — целые числа.
1. Найдите $conv(X)$.
2. Найдите $conv(X \cap \{0, 1\}^n)$.

37. Рассмотрим задачу:

$$\min -x_1 - x_2 - x_3$$

$$x_1 + x_2 \le 1$$

$$x_2 + x_3 \leq 1$$

$$x_1 + x_3 \leq 1$$

$$x_1, x_2, x_3 \geq 0, \text{ целые.}$$

1. Найдите разрыв целочисленности.
2. Постройте правильное неравенство.

5.2. Практические задания

38. Ресторан работает 7 дней в неделю. Повара работают 6 часов в день, 5 дней подряд и затем 2 дня отдыхают. У всех поваров одинаковая зарплата. Приготовление каждого блюда занимает определенное время, и для каждого дня недели установлено общее необходимое количество часов для приготовления пищи. Данные приведены в табл. 8.

    Администратору нужно решить, какое количество поваров нанять и в какие дни они должны работать, чтобы нужное количество часов было отработано, а затраты на оплату труда были минимальными.

Таблица 8

Ежедневное количество рабочих часов

День недели	Требуемое количество часов
понедельник	150
вторник	200
среда	400
четверг	300
пятница	700
суббота	800
воскресенье	300

Постройте математическую модель. Найдите оптимальное решение.

39. Авиакомпания «Альфа» составляет расписание вылетов из Чикаго по следующим направлениям: Колумбия, Денвер, Лос-Анджелес и Нью-Йорк. В каждый город должен состояться ровно один вылет. Вылеты могут быть в 8:00, 10:00 и 12:00. Авиакомпания оплачивает вылет каждого самолета по каждому направлению. Эти затраты составляют 5 тыс. у.е., если вылет совершается до 10:00 включительно, и 3 тыс. у.е. — после 10:00. В каждый момент времени выполняется не более двух рейсов. Кроме того, если в определенное время есть вылет в Нью-Йорк, то в это же время должен быть вылет в Лос-Анджелес. Ожидаемый доход (в тыс. у.е.) от полетов приводится в следующей табл. 9.

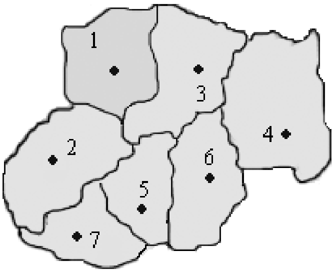

Рис. 25. Расположение районов города

Таблица 9

Ожидаемый доход от полетов

Направление	Время вылета		
	8:00	10:00	12:00
Колумбия	10	6	6
Денвер	9	10	9
Лос-Анджелес	14	11	10
Нью-Йорк	18	15	10

Требуется составить расписание, доставляющее максимальную прибыль авиакомпании. Постройте математическую модель. Найдите оптимальное решение.

40. В городе 7 районов (рис. 25). Известна численность населения в каждом районе (табл. 10).

Таблица 10

Численность населения

Район	Численность населения(тыс. человек)
1	16
2	15
3	98
4	10
5	4
6	6
7	100

В институте по повышению уровня грамотности работает 2 специалиста. В какие районы нужно отправить специалистов, чтобы уровень грамотности в городе стал как можно выше? Если специалист находится в одном из районов, он может посетить еще один соседний район. Районы называются соседними, если их границы на карте города — смежные. Постройте математическую модель. Найдите оптимальное решение.

41. Издательство планирует выпустить новую серию книг по исследованию операций. В серию войдут три книги: OR_1, OR_2, OR_3. Авторы сдают в редакцию рукопись книги, а издательство брошюрует и печатает книгу. В издательстве имеются три машины для печати книг — P_1, P_2, P_3 и две машины для брошюровки — B_1, B_2. Время работы каждой машины с соответствующей книгой указано в табл. 11.

Таблица 11

Время работы машин

	P_1	P_2	P_3	B_1	B_2
OR_1	3	6	4	10	10
OR_2	2	3	3	12	11
OR_3	4	5	5	15	14

Машины могут работать только определенное количество часов. Максимальное время работы (в часах) каждой машины указано в табл. 12.

Таблица 12

Максимальное время работы машин

	P_1	P_2	P_3	B_1	B_2
время работы	120	100	110	$333\frac{1}{3}$	300

Стоимость аренды одной машины не зависит от числа книг и составляет 10, 8, 9, 20 и 23 тыс. у.е. соответственно. Доход от продажи одной книги составляет 40, 60 и 70 у.е. соответственно. Издательство стремится получить максимальную прибыль. При этом оно должно выпустить не менее 500 штук книг OR_3, чтобы поддерживать хороший академический имидж. Постройте математическую модель. Найдите оптимальное решение.

42. Трое рабочих W_1, W_2, W_3 должны выполнить пять работ J_1, J_2, J_3, J_4, J_5. Уровень подготовки и опыт работы у рабочих разный. Время выполнения конкретной работы в часах у каждого работника приводится в табл. 13.

Таблица 13

Время выполнения работы

	J_1	J_2	J_3	J_4	J_5
W_1	5	1	9	4	9
W_2	4	3	8	3	8
W_3	7	5	6	4	7

Каждая работа выполняется рабочим без прерывания. Требуется распределить и выполнить все работы так, чтобы время загрузки всех рабочих было равномерным. Предложите несколько вариантов моделирования равномерной загруженности рабочих (с помощью линейной и квадратичной целевых функций). Найдите оптимальное решение для разных моделей.

43. В лесничестве нужно принять решение, какие участки леса будут вырублены под застройку. Лес разделен на 16 участков прямоугольной формы, как показано в табл. 14.

Таблица 14

Разделение леса на участки

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Соседние участки нельзя выбирать под застройку, чтобы избежать больших вырубленных площадей. Соседними для шестого участка являются, например, участки с номерами 2, 5, 7 и 10. Польза от застройки каждого участка приводится в табл. 15.

Таблица 15

Полезность застройки каждого участка

5	6	2	5
4	3	5	8
8	2	4	9
10	1	6	7

Определите, какие участки выбрать, чтобы общая польза от их застройки была максимальной. Постройте математическую модель, найдите оптимальное решение задачи.

44. Компания «Гига» владеет тремя складами $S1$, $S2$, $S3$ вместимостью 30, 10 и 50 тыс. тонн соответственно. Фиксированные затраты на подготовку к использованию каждого склада составляют 25, 50 и 45 тыс. у.е. соответственно. Первый склад можно расширить за счет использования подземного хранилища на 20 тыс. тонн с дополнительными затратами 1 тыс. у.е. за тонну. При необходимости имеется возможность открыть еще два склада $S4$, $S5$ вместимостью 20 и 30 тыс. тонн с затратами на подготовку 15 и 25 тыс. у.е. соответственно. В настоящий момент в «Гигу» обратились три компании $K1$, $K2$, $K3$ для хранения 20, 60 и 40 тыс. тонн. Компания «Гига» забирает грузы у клиентов и привозит их на склады, транспортные затраты на доставку приводятся в табл. 16.

Таблица 16

Транспортные затраты на доставку

	$K1$	$K2$	$K3$
$S1$	3	7	4
$S2$	2	6	8
$S3$	9	3	4
$S4$	5	6	4
$S5$	7	3	9

Требуется разместить грузы всех клиентов с минимальными суммарными затратами. Нужно ли открывать дополнительные склады? Постройте математическую модель. Найдите оптимальное решение задачи.

45. Импресарио готовит выставку старинных автомобилей, среди которых могут быть Buggati, Cadillac, Cobra, Corvette, Pierce Arrow, Studebaker. Опрос показал, что посмотреть именно Buggati придут 58 специально приглашенных гостей, Cadillac — 37, Cobra — 42, Corvette — 40, Pierce Arrow — 55 и Studebaker — 33. Бюджет организации выставки составляет 15 млн у.е. Стоимость доставки автомобиля на выставку и обеспечение его сохранности составляют 6, 4, 3.8, 4.2, 5.5 и 3.2 млн у.е. соответственно. Задача импресарио в том, чтобы привлечь как можно больше специально приглашенных гостей, не превышая бюджет на организацию. Кроме того, на выставке должно быть не менее трех старинных автомобилей. Если Corvette будет выбран для выставки, то и Cobra должен там быть. Если же Buggati отсутствует, то обязательно нужно включить в показ Cadillac.

Постройте математическую модель и найдите оптимальное решение задачи. Определите, каким может быть минимальный и максимальный бюджет, чтобы выставка состоялась.

46. Компания распространяет технику по пяти городам: Екатеринбург, Омск, Новосибирск, Томск, Иркутск. В настоящий момент 10 снегоуборочных

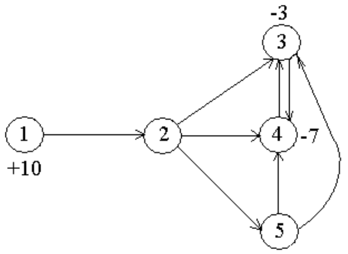

Рис. 26. Сеть дорог

машин находятся в Екатеринбурге и должны быть доставлены в Новосибирск и Томск. В Новосибирск нужно 3 машины, в Томск — 7 машин. Пронумеруем города Екатеринбург, Омск, Новосибирск, Томск и Иркутск целыми числами от 1 до 5 соответственно. Сеть дорог между городами изображена на рис. 26. Вершины сети — города, дуги — дороги между городами. Некоторым вершинам предписан вес — положительное или отрицательное число. Положительное число означает, что в городе соответствующему этой вершине, есть предложение продукции, равное весу вершины, отрицательный вес говорит о том, что в этой вершине имеется спрос на продукцию, соответствующий весу вершины. Предполагается, что сумма весов всех вершин сети равна нулю, это означает, что суммарное предложение совпадает с суммарным спросом.

Таблица 17

Пропускные способности и стоимости перевозок

Дуга	Пропускная способность	Стоимость перевозки
(1,2)	10	11
(2,3)	2	1
(2,4)	3	1
(2,5)	7	2
(3,4)	3	2
(4,3)	2	2
(5,4)	4	5
(5,3)	7	4

Количество водителей, осуществляющих перегон техники из одного города в другой, ограничено, поэтому число снегоуборочных машин, которое

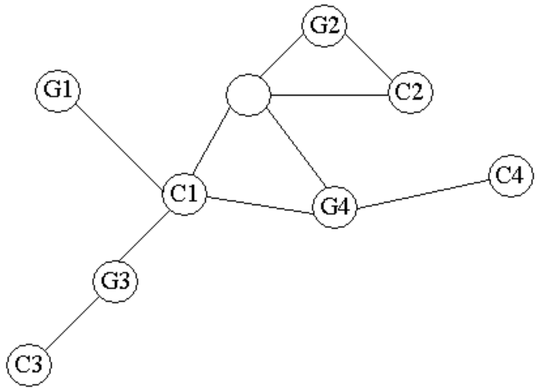

Рис. 27. Электроэнергетическая сеть

можно перевезти из одного города в другой, не должно превышать пропускной способности дуги. Пропускные способности и стоимости перевозок приводятся в табл. 17.

Необходимо составить план перевозок минимальной стоимости так, чтобы удовлетворить спрос и не нарушить пропускных возможностей при перегоне техники.

47. На рис. 27 изображена электроэнергетическая сеть, соединяющая электрогенераторы $G_1, \ldots, G_4$ с источниками потребления $C_1, \ldots, C_4$. Потоки энергии могут идти в любом направлении по ребрам сети, пропускные способности ребер неограничены, стоимость передачи по одному ребру составляет 11 у.е. за 1000 кВт/час. Мощность каждого электрогенератора и стоимость выработки электроэнергии приводятся в табл. 18.

Таблица 18

Данные о выработке электроэнергии

	Электрогенератор			
	G_1	G_2	G_3	G_4
Мощность (тыс. кВт/час)	100	60	80	150
Стоимость выработки (за 1000 кВт/час)	15	13,5	21	23,5

Энергопотребление в $C_1, \ldots, C_4$ составляет 35, 50, 60 и 40 тыс. кВт соответственно. Какие электрогенераторы из сети использовать, чтобы суммарные затраты были минимальные, а потребность в электроэнергии была удовлетворена? Постройте математическую модель и найдите оптимальное решение задачи.

48. Компания производит два вида полимерных материалов: полипропилен и полистирол. Для этого она нанимает неквалифицированных, квалифицированных и высококвалифицированных рабочих. Почасовая оплата в у.е. каждого рабочего и количество тонн материалов, производимых каждым рабочим в час, приводятся в табл. 19.

Таблица 19

Почасовая оплата и почасовые объемы производства

	Неквалифиц.	Квалифиц.	Высококвалифиц.
Оплата	4	6	8
Производительность (тонн в час)			
Полипропилена	2	3	3
Полистирола	0	3	4

Рабочие могут производить одновременно оба продукта, например, квалифицированный рабочий может производить 3 тонны полипропилена и 3 тонны полистирола каждый час. Поступил заказ произвести за один час 21 тонну полипропилена и 15 тонн полистирола.

Требуется определить, сколько рабочих каждой квалификации необходимо нанять, чтобы выполнить заказ с наименьшими затратами на оплату труда. Постройте математическую модель и найдите оптимальное решение.

Компания освоила производство поликарбоната и получила заказ произвести 24 тонны за один час. При этом неквалифицированные рабочие могут произвести 1 тонну поликарбоната в час, квалифицированные рабочие — 3 тонны, а высококвалифицированным рабочим запрещено производить поликарбонат. Как изменится модель при добавлении заказа на новый материал?

Компания получила разрешение на экспорт полистирола. Какую минимальную стоимость продажи за тонну следует установить компании, чтобы запуск производства дополнительных единиц этого материала был целесообразным? Постройте математическую модель. Найдите оптимальное решение задачи.

49. Производитель планирует запуск производства двух новых типов стекла: A и B. Для этого необходимо приобрести специальные печи. Стоимость

печи для производства стекла типа A составляет 500 тыс. у.е., стекла типа B — 600 тыс. у.е.

Для производства стекла необходимы песок, карбонат калия и карбонат кальция в заданных пропорциях (табл. 20).

Таблица 20

Содержание компонент в стекле

	Компоненты		
Тип стекла	Песок	Карбонат калия	Карбонат кальция
A	52 %	13 %	35 %
B	73 %	15 %	12 %

Согласно контракту поставщики могут доставить не более 1460 тонн песка, 500 тонн карбоната калия и 700 тонн карбоната кальция. Доход от продажи 1 тонны стекла типа A составляет 6 тыс. у.е., от продажи 1 тонны стекла типа B — 3 тыс. у.е. Производство устроено таким образом, что если оно запущено, то не менее одной тонны стекла любого типа должно быть произведено.

Постройте математическую модель, максимизирующую прибыль, используя только следующие переменные:

$$y_A(y_B) = \begin{cases} 1, & \text{если покупается печь} \\ & \text{для производства стекла типа A (B)} \\ 0 - & \text{в противном случае,} \end{cases}$$

$x_A(x_B)$ — объем производства стекла типа A (B).

Ответьте на следующие вопросы.

1) Какое из следующих ограничений входит в модель:

1.1) $0.52x_A + 0.73x_b \geq 1460$;

1.2) $0.52x_A + 0.73x_b \leq 14.60$;

1.3) $52x_A + 73x_b \leq 1460$;

1.4) $0.52x_A + 0.73x_b \leq 1460$?

2) Какое из следующих ограничений входит в модель:

2.1) $x_A - 1000y_A \leq 0$;

2.2) $1000x_A - y_A \leq 0$;

2.3) $x_B - 1000y_B \geq 0$?

3) Какое из следующих ограничений входит в модель:

3.1) $x_A - 1460x_A \geq 0$;

3.2) $2000y_A - x_A \geq 0$;

3.3) $1460x_A - y_A \leq 0$;

3.4) $x_A - 1460y_A \leq 0$?

50. Винодел, смешивая четыре сорта винограда, получает три сорта вина. Количество виноградного сока, имеющегося в наличии (литров), пропорции смешивания и стоимость каждого сорта вина указаны в табл. 21.

Таблица 21

Данные о компонентах для производства вина

Сорт вина	Сорт винограда				Стоимость вина (за литр)
	Алиготе	Изабелла	Рислинг	Шардоне	
Каберне	*	$\geq 75\%$ в любой пропорции		$\geq 4\%$	80
Шардоне	*	$\geq 10\%$	*	$\leq 35\%$	50
Мускат	*	$\geq 35\%$		*	35
наличие (в литрах)	130	200	150	350	

$*$ — в любом количестве.

Какие сорта винограда и в каком объеме необходимо смешивать, чтобы получить максимальный доход от продажи вина? Постройте математическую модель и найдите оптимальное решение задачи.

51. В обязанности завхоза входит закупка продуктов питания, расфасованных в жестяные банки. По данным прошлого года, известно необходимое количество банок на каждый месяц (табл. 22).

Таблица 22

Ежемесячное необходимое количество банок

Сен	Окт	Ноя	Дек	Янв	Фев	Мар	Апр	Май
1000	900	850	500	600	1000	1000	1000	500

Завхоз может закупать продукты заранее или в начале каждого месяца по ценам, приведенным в табл. 23.

Таблица 23

Стоимость банки

	Сен	Окт	Ноя	Дек	Янв	Фев	Мар	Апр	Май
Цена (у.е.)	20	20	20	21	21	21	22	22	22

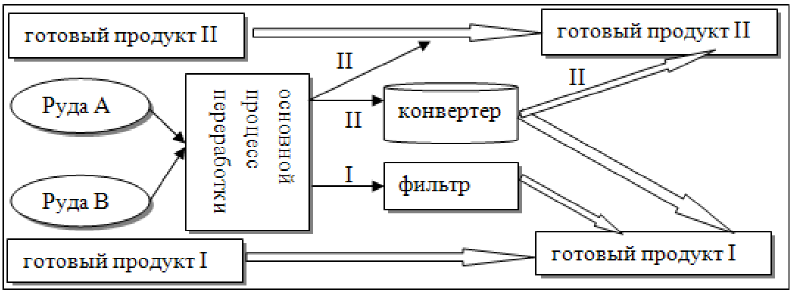

Рис. 28. Процесс переработки 1

Стоимость хранения составляет 0.2 у.е. за одну банку в месяц. В какие месяцы и в каком количестве завхоз должен делать закупки, чтобы обеспечить необходимым количеством продуктов с минимальными затраты на закупку и хранение. Постройте математическую модель и найдите оптимальное решение задачи.

52. Сантехник имеет большой запас 13-метровых медных труб. Ему нужны 10 труб длиной 4 метра, 10 труб длиной 5 метров и 23 трубы длиной 6 метров. Как он должен разрезать 13-метровые трубы? Постройте математическую модель минимизации числа использованных 13-метровых труб. Найдите оптимальное решение задачи.

53. Предприятие занимается переработкой руды. Процесс переработки схематически изображен на рис. 28. Перерабатываются два вида руды: А и В. Предприятию может быть поставлено до 100 тыс. тонн руды вида А по цене 3.25 у.е. за тонну и до 30 тыс. тонн руды вида В по цене 3.40 у.е. за тонну. Общая мощность основного процесса переработки равна 100 тыс. тонн руды при затратах на переработку 0.35 у.е. за тонну. Основной процесс обработки позволяет получать

   - из каждой тонны руды вида А 0.15 тонн продукта I и 0.85 тонн продукта II,
   - из каждой тонны руды вида В 0.25 тонн продукта I и 0.75 тонн продукта II.

Продукт I более ценный, и агрегат, называемый конвертером, способен из каждой тонны продукта II получить 0.5 тонн продукта I и 0.5 тонн продукта II, который нельзя повторно перерабатывать конвертером. Мощность конвертера — 50 тыс. тонн сырья при затратах на конвертерную обработку 0.25 у.е. за тонну сырья. Затраты на фильтрацию продукта I после основного процесса обработки равны 0.10 у.е. тонн сырья.

Условия реализации продукции следующие. Вся продукция идет на продажу. Продукт II может быть реализован в неограниченном количестве по цене 3.80 у.е. за тонну. Продукт I продается по цене 5.50 у.е. за тонну, и его можно продать по этой цене до 45 тыс. тонн. Кроме того, можно продать до 4 тыс. тонн по цене 5.2 у.е. за тонну и неограниченное количество продукта по заниженной цене 5 у.е. за тонну.

Существует контракт, согласно которому требуется поставлять потребителям не менее 40 тыс. тонн продукта I. Оба продукта можно при необходимости докупить: закупочная цена продукта I равна 5.75 у.е. за тонну, закупочная цена продукта II — 4 у.е. за тонну.

Постройте математическую модель, выполните следующие задания:

1) найдите план выпуска продукции с максимальной прибылью;
2) покажите, как изменится оптимальный план, если мощность фильтра ограничена величиной 15 тыс. тонн;
3) покажите, как изменится оптимальный план производства, если фильтр сломается и не будет задействован в производственном процессе, а продукт I в этом случае будет продаваться только по цене 5 у.е. за тонну?

Постройте математическую модель для задачи, в которой требуется найти единую (не зависящую от объемов продаж) минимальную цену на продукт I, при которой прибыль предприятия будет не меньше 50 тыс. у.е. По-прежнему ли это модель линейного программирования?

54. Выполните упражнение 53 со схемой работы предприятия, изображенной на рис. 29.

55. Выполните упражнение 53 со схемой работы предприятия, изображенной на рис. 30. Затраты на упаковку продукта I составляют 0.15 у.е. за тонну сырья.

56. На складе имеются прямоугольные листы материала размером 100×100 см^2 в количестве 100 штук (рис. 31) и деловые отходы в форме многоугольника в количестве 70 штук (рис. 32). В цех поступил заказ на изготовление деталей форм, указанных на рис. 33 в следующих количествах: невыпуклых многоугольников — 550 штук; четвертей круга радиусом 30 см — 800 штук; равнобедренных прямоугольных треугольников с

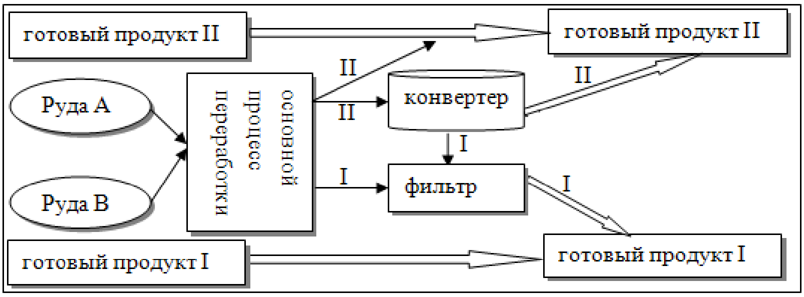

Рис. 29. Процесс переработки 2

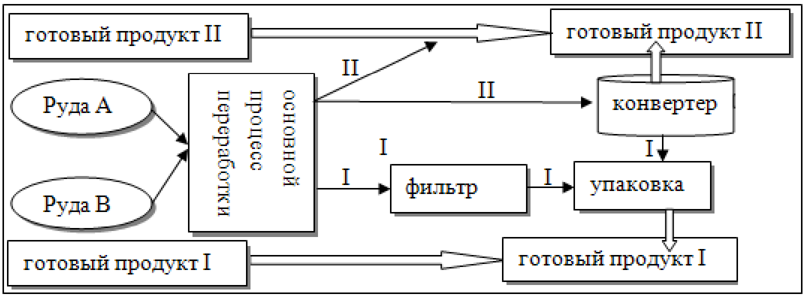

Рис. 30. Процесс переработки 3

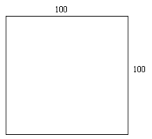

Рис. 31. Заготовки

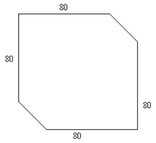

Рис. 32. Деловые отходы

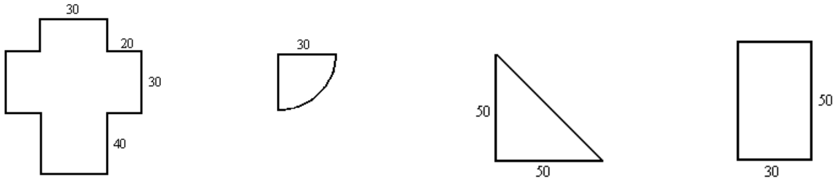

Рис. 33. Формы деталей в заказе

катетами 50 см — 330 штук; прямоугольников 50×30 см^2 — 700 штук. При раскрое материала разрешаются произвольные повороты деталей. Найдите, какое минимальное число листов материала нужно дополнительно закупить, чтобы выполнить заказ? Постройте математическую модель, решите и предъявите использованные карты раскроя.

57. Завод может выпускать насосы 10 типов: $V1$, $V2$, $SV1$, $SV2$, $SV3$, $W1$, $W15$, $W2$, $SW2$, $SW3$. Для производства насосов каждого типа известны:
— затраты (в тыс.у.е.) на наладку оборудования (табл. 24),

Таблица 24

Затраты на наладку оборудования

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
1	1	1.5	1.5	1.5	1.2	1.3	1.9	2	5

— удельная стоимость (в у.е.) производства (табл. 25).

Таблица 25

Удельная стоимость производства

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
100	110	120	120	140	110	115	120	110	200

Насосы обладают свойством заменяемости. В каждом столбце j (табл. 26) в строке i указывается количество насосов типа i, способных заменить один насос типа j.

Таблица 26

Коэффициенты заменяемости насосов

	V1	V2	SV1	SV2	SV3	W1	W15	W2	SW2	SW3
V1	1	**2**	3	-	-	2	3	-	-	-
V2	1	1	2	5	-	-	-	-	-	-
SV1	-	-	1	4	10	1	2	-	-	-
SV2	-	5	1	1	5	-	-	-	2	4
SV3	-	-	1	1	1	-	-	-	2	2
W1	1	-	1	-	-	1	1	2	-	-
W15	1	-	1	-	-	-	1	2	-	-
W2	-	-	-	-	-	1	1	1	2	-
SW2	-	2	-	1	-	-	-	1	1	2
SW3	-	1	-	-	1	-	1	1	1	1

символ «—» означает, что замена невозможна.

Например, выделенное жирным шрифтом число 2 в табл. 57 означает, что вместо одного насоса $V2$ можно произвести 2 насоса $V1$.

Решение о том, какие типы насосов будут заменены, принимается один раз до запуска производства. Каждый тип насоса можно заменить в соответствующей пропорции, указанной в таблице, только на один.

Требуется найти минимальный по затратам план выпуска насосов для выполнения заказа, указанного в табл. 27.

Таблица 27

Заказ

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
0	29	0	10	0	0	15	20	0	0

Постройте математическую модель, решите задачу и ответьте на следующие вопросы.

1) Как изменится план выпуска насосов, если суммарные затраты на наладку оборудования не должны превышать 3000 у.е.?
2) Изменится ли номенклатура выпускаемых насосов, если на складе имеется запас насосов типа $SW3$ в количестве 50 шт.?
3) Известно, что для производства одного насоса требуется определенное количество килограмм резины указанное в табл. 28.

Таблица 28

Необходимое количество килограмм резины

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
1	2	1	3	1	5	1	8	2	7

Завод обладает запасом резины в размере 10 кг. Как изменится план выпуска, если завод может закупать резину по цене 20 у.е. за один килограмм?

4) Постройте математическую модель для случая, когда любой тип насосов можно заменять не более двумя (тремя) другими допустимыми типами.

58. Завод может выпускать насосы 10 типов: $V1$, $V2$, $SV1$, $SV2$, $SV3$, $W1$, $W15$, $W2$, $SW2$, $SW3$. Время на наладку оборудования (в тыс. мин.) приводится в табл. 29.

Таблица 29

Время на наладку оборудования

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
1	1	1.2	1.2	1.4	1.1	1.15	1.2	1.1	2

Время, затрачиваемое на производство одного насоса в часах приводится в табл. 30.

Таблица 30

Время на производство

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
100	110	120	120	140	110	115	120	150	150

Насосы обладают свойством заменяемости. Условия заменяемости описаны в упражнении 57. Наладка оборудования для разных типов насосов начинается одновременно, начало производства разных типов насосов может осуществляться параллельно.

Требуется найти оптимальный по времени план выпуска насосов для выполнения заказа, указанного в табл. 31.

Таблица 31

Заказ насосов

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
0	29	0	10	0	0	15	20	0	0

Постройте математическую модель, решите задачу и ответьте на следующие вопросы.

1) Как изменится план выпуска насосов, если суммарное время на наладку оборудования не должно превышать 3500 часов?
2) Изменится ли номенклатура выпускаемых насосов, если на складе имеется запас насосов типа $SW3$ в количестве 50 шт.?
3) Известно, что наладка оборудования для производства каждого типа насосов требует финансовых затрат (тыс. у.е.), указанных в табл. 32.

Таблица 32

Финансовые затраты на наладку оборудования

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
1	1	1.5	1.5	1.5	1.2	1.3	1.9	2	5

Удельная себестоимость (в у.е.) производства насосов каждого типа приводится в табл. 33.

Таблица 33

Удельная себестоимость производства насоса

$V1$	$V2$	$SV1$	$SV2$	$SV3$	$W1$	$W15$	$W2$	$SW2$	$SW3$
100	110	120	120	140	110	115	120	110	200

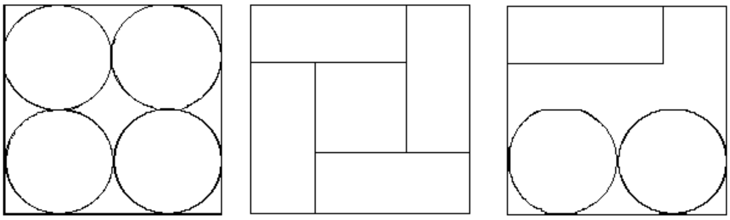

Рис. 34. Образцы раскроя

Как изменится план выпуска, если суммарные затраты на выполнение заказа не должны превышать 10000 у.е., а суммарное время на наладку оборудования не должно превышать 5000 часов?

59. Строительный магазин столкнулся со следующей проблемой. Клиентам нужен оргалит размером 120×90 см^2, 120×150 см^2 и 120×180 см^2 в количестве не менее 20, 50 и 40 листов по цене 700, 900 и 1000 у.е. за один лист соответствующего размера. Магазин может вырезать листы нужного размера из стандартных листов большего размера 120×240 см^2, которых имеется неограниченное количество, оптовая стоимость закупки таких листов составляет 600 у.е. за один лист. Стоимость выполнения одного разреза составляет 150 у.е.

   Каким образом следует разрезать большие листы, чтобы получить максимальную прибыль? Постройте математическую модель, решите задачу и предъявите схемы разрезов.

60. У раскройщика имеется 30 листов фанеры размера 300×300 см^2, из которых необходимо выкроить 20 кругов диаметром 150 см и 15 прямоугольников размером 120×180 см^2. Образцы раскроя приводятся на рис. 34. Стоимость выкраивания одного круга составляет 200 у.е., одного прямоугольника — 150 у.е. В стоимость включено выполнение всех разрезов, и она не зависит от выбранного образца. Круги и прямоугольники можно купить готовые по цене 450 и 300 у.е. соответственно, если не хватит больших листов. Раскройщик стремится минимизировать расходы на выкраивание. При этом он должен получить нужное количество кругов и прямоугольников и использовать больших листов не больше, чем у него есть, а суммарные отходы должны быть не более 30 %. Постройте математическую модель и найдите решение.

61. Менеджер компании «K&Co» должен выбрать, в какие из пяти проектов A, B, C, D или E сделать инвестиции на ближайшие 3 года так, чтобы получить максимальную прибыль. Проекты могут принести доход (млн у.е.), указанный в табл. 34.

Таблица 34

Доход от проектов

A	B	C	D	E
3	2	1	4	2

Менеджер обратился к аналитикам и получил следующие рекомендации.

1) По крайней мере в один из проектов C, D или E обязательно нужно сделать инвестиции, так как они идеально соответствуют специализации сотрудников компании.
2) Если для инвестиций выбран проект A, то и проект D тоже должен быть выбран.
3) Если выбраны проекты A и B, то в проект C не нужно инвестировать из-за высоких рисков.
4) Если выбраны проекты B и C, то нужно инвестировать и в E, поскольку в этом случае можно получить хорошие скидки от поставщиков.

Постройте математическую модель и найдите оптимальное решение задачи.

62. В 2015 году Алиса планирует отправиться в кругосветное путешествие. В начале 2011 года она готова начать делать сбережения для путешествия так, чтобы к началу 2015 года у нее было 21 тыс. евро.

Она может выбирать из трех типов акций A, B или C. Каждая акция стоит 100 у.е. Покупка акций осуществляется в начале каждого года. Через год инвестиций в акции типа A выплачивается 104 у.е., через два года инвестиций в акции типа B — 110 у.е., через четыре года инвестиций в акции типа C — 125 у.е. В начале каждого года — 2011, 2012, 2013 и 2014 Алиса планирует инвестировать всего не более 5 тыс. у.е.

Необходимо найти минимальный суммарный объем инвестиций за 2011–2014 годы. Постройте математическую модель и найдите оптимальный план.

Как изменится модель, если появляется еще один тип инвестиций: акции типа D? Стоимость каждой такой акции составляет 100 у.е., а через три года ее стоимость составит 115 у.е.

63. Завод производит некоторую продукцию. Клиенты сразу забирают её с места производства, так как у завода нет складских помещений. Если же объемы производства превысили спрос, то продукция отправляется на склад магазина. Если же произведенной продукции недостаточно для удовлетворения спроса, то недостающее количество нужно привести со склада магазина. Продукция отправляется на склад или забирается со склада в конце дня, а поступает на склад или к клиентам в начале следующего дня. Удельные транспортные затраты на перевозку продукции с или на склад составляют 0.2 у.е.

    Известно, что в первый день на складе лежит 100 единиц продукции, к концу 4-го дня там должно оставаться 100 единиц продукции. Ежедневный спрос и максимальный объем производства в единицах приводятся в табл. 35.

Таблица 35

Ежедневный спрос и максимальный объем производства

День	Спрос	Производство
первый	60	70
второй	80	50
третий	70	90
четвертый	50	70

    Удельные затраты на производство составляют 5 у.е. в день, удельная стоимость хранения — 0.1 у.е. в день.

    Необходимо составить план производства, перевозки и управления запасами продукции, чтобы удовлетворить спрос на 4 дня с минимальными суммарными затратами. Постройте математическую модель и найдите оптимальный план инвестиций.

64. Фирма специализируется на производстве высококачественных горных велосипедов одной модели. Спрос велосипедов в каждом месяце ограничен. Каждый раз, чтобы запустить производство, необходимо оплатить наладку оборудования. Производить можно не более одной партии в месяц. Предполагается, что производственные мощности фирмы неограничены.

    Затраты на наладку производства составляют 5000 у.е., стоимость производства одной единицы составляет 100 у.е. Таким образом, производство партии из 1 велосипеда требует затрат в 5100 у.е., для производства партии из 10 велосипедов затраты составят $5000 + (10 \cdot 100) = 6000$ у.е.

    В табл. 36 приводится прогноз ежемесячного спроса d_t на велосипеды на следующий год.

Таблица 36

Ежемесячный спрос

Янв	Фев	Мар	Апр	Май	Июн
400	400	800	800	1200	1200
Июл	Авг	Сен	Окт	Ноя	Дек
1200	1200	800	800	400	400

Известно, что на складе осталось 200 велосипедов с прошлого года. Удельная стоимость хранения в месяц составляет 5 у.е., вместимость склада неограничена. Задача менеджера — составить такой план производства и хранения велосипедов, чтобы удовлетворить спрос с минимальными суммарными затратами на год.

65. Проект состоит из 8 работ. Длительности выполнения каждой работы (дней) приводятся в табл. 37.

Таблица 37

Длительность выполнения работы

Номер работы	1	2	3	4	5	6	7	8
Длительность	3	2	4	3	2	1	5	4

Работы можно выполнять на несколько дней быстрее, но не быстрее, чем на максимальное количество дней, приведенное в табл. 38.

Таблица 38

Максимальное сокращение дней

Номер работы	1	2	3	4	5	6	7	8
Кол-во дней	1	1	2	1	1	1	2	1

Стоимость ускорения выполнения работы на один день (в у.е.) приводится в табл. 39.

Таблица 39

Стоимость ускорения выполнения работы

номер работы	1	2	3	4	5	6	7	8
стоимость	10	10	12	11	11	12	13	13

У некоторых работ имеются непосредственные работы-предшественники (табл. 40).

Таблица 40

Отношения предшествования

Номер раб.	1	2	3	4	5	6	7	8
Номер раб.-предшест.	—	—	1,2	1,2	3,4	3,4	3	5,6

Постройте сеть (работы – дуги) и выберите правильные варианты ответов на следующие вопросы.

Наиболее раннее время завершения всего проекта равно

1) 11;
2) 12;
3) 13;
4) 14.

Наиболее позднее время завершения работы 4 равно

1) 6;
2) 7;
3) 8;
4) 9.

Какой путь является критическим в данном проекте:

1) 1-3-7;
2) 1-3-6-8;
3) 1-4-5-8;
4) 1-3-5-8?

Постройте математическую модель и найдите минимальные суммарные затраты при условии, что проект должен быть завершен за 10 дней.

66. На прошлой неделе Ксения, Алексей, Степа, Наталья и я каждый вечер с понедельника по пятницу ужинали вместе. Каждый из нас в начале каждого дня по очереди выбирал подходящее место. В пятницу Наталья пропустила ужин с друзьями, так как навещала родственников, но, тем не менее, выбрала подходящее место для друзей. Мы посетили следующие места: французский ресторан, суши-бар, пиццерию, греческий ресторан и пивоварню. Ксения повела нас в среду. Пятничный ужин был в пивоварне. Степа не любит суши, и он первый выбирал место для ужина. Я выбрала французский ресторан, а на следующий день один из друзей повел всех в пиццерию. Ответьте на вопрос, кто, когда и какое место для ужина выбрал? (Используйте задачу о выполнимости.)

67. Пять парикмахеров решили подстричь друг друга. В табл. 41 приводится время в минутах, затрачиваемое на стрижку.

Таблица 41

Продолжительность стрижки

	Елена	Максим	Наталья	Ирина	Ольга
Елена	–	42	37	36	44
Максим	31	–	43	48	45
Наталья	43	34	–	27	44
Ирина	26	28	44	–	40
Ольга	47	43	42	31	–

Составьте математическую модель и найдите назначения между парикмахерами так, чтобы подстричь всех парикмахеров за наименьшее общее время.

68. На склад экспресс-почты прибыло восемь посылок, которые требуется доставить восьми адресатам. За хранение каждой посылки в течение одного часа взимается плата в размере c_i у.е., время доставки посылки i до адресата составляет t_i часов, $i = 1, \ldots, 8$, данные приводятся в табл. 42.

Таблица 42

Время доставки посылок

	1	2	3	4	5	6	7	8
c_i	8	8	8	4	5	8	7	2
t_i	11	14	10	16	19	18	16	19

В службе доставки работает только один курьер. Постройте математическую модель и найдите, в какой последовательности нужно доставить посылки, чтобы суммарная стоимость хранения была минимальной.

69. Завод специализируется на производстве продукции из стекла. Установлено три производственных линии для производства шести типов продуктов, которые нужно произвести за 15 дней. На завод поступает заказ на разные типы продуктов. Необходимо составить план выполнения заказа с минимальными затратами на производство и хранение продукции.

Единовременно каждая линия может производить только один тип продукта. Каждый тип продукта в каждый момент времени производится только на одной линии. Операции на линии выполняются целое число дней. Линия работает непрерывно без остановок. Переход с производства одного продукта на другой для каждой линии осуществляется в начале дня. При этом некоторое время, пока оборудование перенастраивается, производственные мощности падают. В это время производится продукт низкого качества, который потом уходит в отходы. Объемы некачественного продукта зависят от типов продуктов, между которыми был переход, и от производственной линии. Скорость производства каждой линии постоянна и не зависит ни от типа продукта, ни от величины заказа на него.

Суммарные затраты складываются из затрат на хранение продукции и производственных затрат. Известны ежедневные производственные затраты на каждый тип продукта на любой линии. В начале периода планирования готовой продукции нет.

В день можно произвести не более 342 единиц каждого типа продукта на каждой линии. Ежедневная удельная стоимость хранения (в у.е.) каждого типа продукта приводится в табл. 43.

Таблица 43

Ежедневная удельная стоимость хранения

Продукт	1	2	3	4	5	6
Стоимость хранения	5	5	1	3	2	1

Ежедневная удельная стоимость производства (в у.е.) каждого продукта на любой линии приводится в табл. 44.

Таблица 44

Ежедневная удельная стоимость производства

Продукт	1	2	3	4	5	6
Стоимость производства	800	800	600	400	400	800

Таблица 45

Ежедневный спрос на каждый тип продукта

День	Величина спроса на каждый продукт					
	1	2	3	4	5	6
1	0	85	0	102	0	89
2	76	0	77	0	111	0
3	116	95	124	92	87	0
4	102	0	106	0	94	83
5	109	101	79	118	91	86
6	100	120	123	108	90	118
7	93	99	107	95	111	108
8	98	120	85	78	106	83
9	93	114	98	108	117	79
10	118	94	105	84	105	80
11	83	78	119	111	91	99
12	122	91	98	121	97	92
13	77	86	125	122	107	98
14	123	104	114	102	83	78
15	125	97	109	80	103	95

Количество единиц продукции j-го типа низкого качества при переключении с i-го продукта на k-й линии рассчитывается по формуле $c_k^{ij} = st^i + 10(k-1) + 10|j-i|$, где $st = (40, 20, 20, 20, 40, 50)$ — начальное количество низкокачественного продукта. После переключения линии на новый продукт (пока производится низкокачественый) производственные затраты считаются для нового продукта. В начале горизонта планирования все линии настроены на производство качественного продукта. Данные о спросе на каждый тип продукта на каждый день приводятся в табл. 45. Постройте математическую модель и найдите оптимальное решение задачи.

70. Предприятие установило систему производственного планирования в своих отделениях. Задача системы — осуществлять планирование производства и хранения продукта. Производственный процесс состоит из нескольких шагов: переработка сырья, упаковка продукта, хранение и отправка упакованного продукта потребителям. Планирование осуществляется на 3 месяца.

    Схема производственного процесса представлена на рис. 35. Процесс производства начинается с переработки сырья, количество которого неограничено. В результате получается 4 типа продукта: A, B, C, D. Ежемесячно каждого продукта можно произвести не более C_i штук, $i = A,\ B,\ C,\ D$. Известна потребность D_i^t штук i-го типа продукта в месяц t. После переработки сырья продукт отправляется либо в цех по упаковке, либо на склад. Вместимость склада неограничена. Цех по упаковке может обрабатывать не более 28000 штук продукции в месяц. Упакованный продукт может отправляться либо на склад, либо к потребителям. На складе имеются

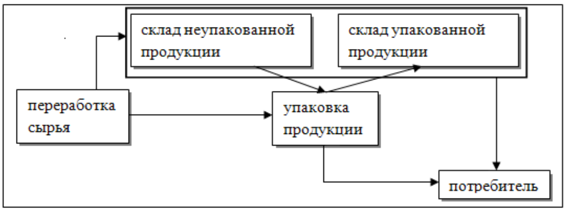

Рис. 35. Схема производственного процесса

помещения для хранения упакованной и неупакованной продукции. Поскольку для хранения упакованной продукции нужны специальные стеллажи, то затраты на хранение упакованного продукта на 4 у.е. за штуку в месяц больше, чем затраты на хранение неупакованного продукта. Данные приводятся в табл. 46.

Таблица 46

Затраты на хранение неупакованного продукта

	D_i^t (шт.)				
i	$t=1$	$t=2$	$t=3$	C_i (шт. в мес.)	Затраты на хранение неупакованного продукта (у.е. за шт. в мес.)
A	5000	6000	3000	8000	35
B	900	1000	4000	5000	39
C	6000	9000	4000	8000	45
D	10000	11000	14000	15000	85

1) Какое количество продукции необходимо производить ежемесячно, чтобы удовлетворить запросы потребителей с наименьшими суммарными затратами?

2) Стоит ли увеличивать мощность цеха по упаковке? Какие данные Вам необходимы, чтобы ответить на этот вопрос?

3) При смене продукта необходимо очищать упаковочную линию, а для этого используются дорогостоящие препараты. В начале каждого месяца, кроме первого, линию нужно очищать, если тип продукта меняется. Затраты на очистку упаковочной линии перед упаковкой каждого типа продукта приводятся в табл. 47.

Таблица 47

Затраты на очистку упаковочной линии

Продукт	A	B	C	D
Стоимость очистки (тыс. у.е.)	500	900	800	900

Найдите новый план производства и хранения продукции, доставляющий минимальные суммарные затраты. Постройте математическую модель и найдите оптимальное решение.

71. Промышленное предприятие получило разрешение на разработку месторождения руды на участке размером 200×200 футов^2. Вскрытие участка производится наклонными выработками, предельный уклон наклона 45^0. С учетом максимального угла наклона и с целью предотвращения засыпания нижних слоев и заполнения всего поперечного сечения выработки выемку руды делают блоками высотой 25 футов, шириной и длиной по 50 футов, размещая их, как показано на рис. 36. Чтобы добраться до внутреннего блока, не лежащего на поверхности, нужно отработать «пирамиду» из четырех блоков, лежащих на один слой выше этого блока. Проведенные исследования местности позволили оценить возможное со-

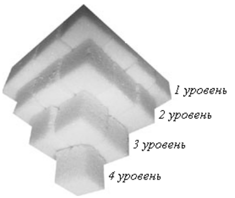

Рис. 36. Выработка внутреннего блока

держание руды (в процентах) в разных местах участка на разной глубине для каждого блока (табл. 48—51).

Таблица 48

Содержание руды на 1 уровне

1 уровень (поверхность)	1.5	1.5	1.5	0.75
	1.5	2.0	1.5	0.75
	1.0	1.0	0.75	0.5
	0.75	0.75	0.5	0.25

Таблица 49

Содержание руды на 2 уровне

2 уровень (глубина 25 футов)	4.0	4.0	2.0
	3.0	3.0	1.0
	2.0	2.0	0.5

Таблица 50

Содержание руды на 3 уровне

3 уровень (глубина 50 футов)	12.0	6.0
	5.0	4.0

Таблица 51

Содержание руды на 4 уровне

4 уровень (глубина 75 футов)	6.0

Если бы блок состоял на 100 % из руды, то он приносил бы доход в 200 тыс. у.е.

Стоимость выработки одного блока зависит от глубины (табл. 52).

Таблица 52

Стоимость выработки одного блока на разной глубине

Уровень	Стоимость (тыс. у.е.)
1	3
2	6
3	8
4	10

Какие блоки следует вырабатывать, чтобы получить максимальную прибыль? Постройте математическую модель и найдите оптимальное решение. В модели должно быть 56 ограничений и 30 переменных.

Можно ли использовать методы решения задач линейного программирования для поиска оптимального решения?

72. Предприятие занимается переработкой неочищенных масел. Перед тем, как получить готовый продукт, сырье проходит две стадии обработки. Сначала масла очищаются, а затем смешиваются. В наличии имеется пять типов масел: два из которых растительного происхождения $V1$, $V2$, три — животного $O1$, $O2$, $O3$. Горизонт планирования производства — с января по июнь. Сырье можно покупать как непосредственно перед использованием в любой месяц горизонта планирования, так и заранее. В табл. 53 приводится стоимость покупки тонны масла в соответствующий месяц.

Таблица 53

Стоимость тонны масла

	$V1$	$V2$	$O1$	$O2$	$O3$
Январь	110	120	130	110	115
Февраль	130	130	110	90	115
Март	110	140	130	100	95
Апрель	120	110	120	120	125
Май	100	120	150	110	105
Июнь	90	100	140	80	135

Конечный продукт продается по цене 150 у.е. за тонну.

Очистка растительных и животных масел выполняется на разных аппаратах. Согласно технологическим требованиям аппараты ежемесячно могут очищать не более 200 тонн растительного масла, и не более 250 тонн масла животного происхождения. В процессе очистки сырье не теряет в весе. Затратами на очистку можно пренебречь.

На складе можно хранить всего до 1000 тонн неочищенного растительного и столько же животного происхождения масел. Стоимость хранения любого масла 5 у.е. за тонну в месяц. Масло, прошедшее очистку, и готовый продукт не могут храниться на складе. В начале января на складе лежит 500 тонн масла каждого происхождения и столько же должно остаться к концу горизонта планирования, т. е. в конце июня.

Качество готового продукта оценивается его плотностью. Этот показатель зависит прямо пропорционально от массы (и только от нее) и должен принимать значение от 3 до 6. Плотность масел приводится в табл. 54. Постройте математическую модель и определите, какие масла и в каком количестве должен покупать производитель, чтобы получить максимальную прибыль.

Таблица 54

Плотность масел

$V1$	$V2$	$O1$	$O2$	$O3$
8.8	6.1	2.0	4.2	5.0

73. Модифицируйте модель для задачи 72, учитывая следующие требования.
1. При изготовлении продукта в смеси всегда должно присутствовать не более трех различных типов масел.
2. Нельзя использовать меньше 20 тонн масла любого типа в месяц.
3. Если при изготовлении продукта в смеси присутствует масло $V1$ или $V2$, то обязательно должно быть и масло $O3$.

74. Крупная компания собирается принять решение о переносе некоторых филиалов из Лондона в другие города. Это позволит сотрудникам компании приобрести недорогое жилье, обеспечить рабочими местами жителей других городов. Новые филиалы могут размещаться в Эдинбурге, в Брайтоне либо остаться в Лондоне. Необходимо разместить пять филиалов. В каждом городе может быть расположено не более трех филиалов. Выгода (тыс. у.е. в год) от перемещения филиалов в другие города приводится в табл. 55.

Таблица 55

Выгода от перемещения филиалов

	A	B	C	D	E
Эдинбург	10	15	10	20	5
Брайтон	10	20	15	15	15

При размещении в разных городах возникают затраты на сообщение между филиалами, которые зависят от местоположения филиалов и объемов продукции, пересылаемых из одного филиала в другой. В табл. 56 приводятся объемы (тыс. единиц) пересылаемые, между филиалами.

Таблица 56

Объемы пересылки между филиалами

	A	B	C	D	E
A	0	0	1	1.5	0
B	0	0	1.4	1.2	0
C	0	0	0	0	2
D	0	0	0	0	0.7

В табл. 57 приводятся стоимости пересылки единицы продукции в год (у.е.).

Таблица 57

Удельная стоимость пересылки

	Эдинбург	Брайтон	Лондон
Эдинбург	5	14	13
Брайтон	0	5	9
Лондон	0	0	10

Где нужно разместить филиалы, чтобы минимизировать ежегодные общие расходы? Постройте математическую модель целочисленного линейного программирования и найдите оптимальное решение.

75. Крупная кинокомпания планирует съемку фильма. Известно, что в съемке фильма будут участвовать n актеров. Фильм состоит из m эпизодов. Каждый день снимается ровно один эпизод. Для каждого эпизода известны актеры, которые в нем задействованы. Актер может сниматься в нескольких эпизодах. Он приезжает на съемочную площадку в тот день, когда снимается первый эпизод с его участием, и покидает площадку в конце последнего дня съемок всех эпизодов с его участием. За каждый день присутствия на съемочной площадке (независимо от того, занят ли актер в съемке в этот день или нет) кинокомпания выплачивает ему персональный гонорар.

Найдите, в каком порядке должны сниматься эпизоды, чтобы сумма выплаченных гонораров была минимальной.

Постройте математическую модель и найдите оптимальное решение для съемок фильма «Новосибирск, я тебя люблю!».
Эпизоды фильма:
1. Знакомство с городом. Участвуют: Брюс, Джеки, Брэд, Том и Пенелопа.
2. У фонтана. Участвуют: Брэд, Джулия и Сальма.
3. В парке. Участвуют: Брэд, Том и Джулия.
4. Знакомство с университетом. Участвуют: Брюс, Джеки и Пенелопа.
5. Прогулка. Участвуют: Брэд и Сальма.
6. В театре. Участвуют: Брюс, Брэд, Том, Джулия и Пенелопа.
Персональные гонорары актеров (в тыс. у.е. за один день) приводятся в табл. 58.

Таблица 58

Персональные гонорары актеров

Брюс	Джеки	Брэд	Том	Пенелопа	Джулия	Сальма
100	70	65	75	110	130	120

6. Решение оптимизационных задач в GAMS

В этом разделе на одном из примеров, рассмотренном в начале пособия, будет разобран процесс построения модели в системе GAMS.

GAMS — это аббревиатура английских слов General Algebraic Modeling System, что можно перевести как «система для алгебраического моделирования». Система GAMS представляет собой оболочку для описания моделей. Она не решает задачу, а направляет к одной из решающих программ (solver или решатель) CPLEX, BARON и др.

Пример

Фирма производит шесть типов стульев: *офисный*, *помощник*, *капитан*, *маркиза*, *испанский* и *венский*. Для производства стульев необходимы следующие детали: длинные и короткие болты, тяжелые и легкие сиденья, длинные и короткие ножки, перекладины и др. В табл. 59 приводятся данные о стоимости стульев, потребности в деталях для каждого типа стула и их наличие на складе.

Таблица 59

Количество деталей на складе

Тип стула	*Офис*	*Пом*	*Кап*	*Марк*	*Исп*	*Вен*	
Цена	36	40	45	38	35	25	
Требуемое количество деталей							Всего на складе
длин. болты	8	0	12	0	8	4	1280
корот. болты	4	12	0	12	4	8	1900
тяж. сиденья	4	4	4	4	4	4	1090
лег. сиденья	1	0	0	0	1	1	190
длин. ножки	0	1	1	1	0	0	170
корот. ножки	6	0	4	0	5	0	1000
перекладины	0	4	0	5	0	6	1000
гайки	1	0	0	0	0	0	110
роллеры	0	1	0	0	0	0	72
каркас	0	0	1	1	0	0	93
крепления	0	0	0	0	1	1	85

Необходимо определить, какие стулья и в каком количестве нужно производить, чтобы получить максимальный доход, используя имеющееся количество деталей на складе. Введем обозначения:

I — множество типов стульев, $I = \{$*кап.*, *пом.*, *мар.*, *исп.*, *вен.*, *офис.*$\}$;

J — множество деталей, $J = \{$*дл.бол.*, *кор.бол.*, *тяж.сид.*, *лег.сид.*, *длин.нож.*, *кор.нож.*, *перек.*,*гайк.*, *ролл.*, *кар.*, *креп.*$\}$;

p_i — цена i-го типа стула, $i \in I$;

d_{ji} — количество деталей j-го типа, необходимое для производства одного стула i-го типа, $i \in I$, $j \in J$;

q_j — запас деталей j-го типа на складе, $j \in J$.

Введем переменные:

x_i — количество стульев i-го типа, которое производит фирма;
z — суммарный доход от продажи стульев. С точки зрения моделей целочисленного линейного программирования необходимости в такой переменной нет. Она необходима для работы с моделью в GAMS. Построим математическую модель:

$$\max\ z$$

$$z = \sum_{i \in I} p_i x_i \qquad \textit{function}$$

$$\sum_{i \in I} d_{ji} x_i \le q_j, j \in J \qquad \textit{constraints1}$$

$$x_i \ge 0, \text{целые}, i \in I \qquad \textit{constraints2}$$

Целевая функция — максимизировать суммарный доход (*function*). Группа ограничений *constraints1* гарантирует, что имеющихся деталей на складе достаточно для производства стульев. Группа ограничений *constraints2* определяет допустимые значения переменных.

Ниже приводится текст программы для решения этой задачи в GAMS.

```
$ontext
Решается задача о работе мебельного комбината.
Математическая модель записана
в терминах целочисленного линейного программирования.
$offtext
Sets
   I множество стульев /1*6/
   J множество деталей /1*11/
;
Parameters
p(i) стоимость i-го типа стула i из I
  /1 36
   2 40
   3 45
   4 38
   5 35
   6 25 /
q(j) кол-во деталей j-го типа на складе j из J
 /1 1280
  2 1900
  3 1090
  4 190
  5 170
```

```
24.   6 1000
25.   7 1000
26.   8 110
27.   9 72
28.  10 93
29.  11 85 /;
30. Table
31. d(j,i) кол-во деталей j-го типа для пр-ва стула i-го типа
32.      1  2   3   4   5  6
33.  1  8  0   12  0   8  4
34.  2  4  12  0   12  4  8
35.  3  4  4   4   4   4  4
36.  4  1  0   0   0   1  1
37.  5  0  1   1   1   0  0
38.  6  6  0   4   0   5  0
39.  7  0  4   0   5   0  6
40.  8  1  0   0   0   0  0
41.  9  0  1   0   0   0  0
42. 10  0  0   1   1   0  0
43. 11  0  0   0   0   1  1 ;
44. Integer Variables
45. x(i)сколько стульев $i$-го типа производит фирма;
46. x.up(i)=INF;
47. Free Variable
48. z суммарный доход от продажи стульев;
49. Equations
50. function  суммарный доход от продажи
51. constraints1(j) ограничения на кол-во деталей на складе;
52. function.. z =e= sum(i,x(i)*p(i));
53. constraints1(j).. sum(i,d(j,i)*x(i)) =l= q(j);
54. Model Chairs /function,constraints1/;
55. *Model Chairs /all/;
56. Chairs.optcr=0.0;
57. Solve Chairs using mip max z;
58. Display x.l, z.l, constraints1.l;
```

Программа состоит из трех частей. Первая часть — описание исходных данных задачи, вторая часть — описание переменных, декларирование ограничений и непосредственное описание модели (целевой функции и ограничений), третья часть — обращение к решателю и выдача результатов.

Строки 1—5 соответствуют комментариям. Для того, чтобы закомментировать одну строку, нужно поставить знак «*» в начале строки. Для того, чтобы закомментировать несколько строк, нужно окружить их командами $\$ontext$ и $\$offtext$.

Строки 6—42 соответствуют описанию исходных данных задачи. Для этого используются специальные блоки: $Sets$ — для задания множеств, $Parameters$ — для векторных величин, $Tables$ — для матриц, $Scalars$ — для скалярных величин и констант. Эти блоки могут следовать в произвольном порядке друг за другом, но должны при этом использовать определенные ранее величины.

Для сокращения кода программы типы стульев *капитан*, *помощник*, *маркиза*, *испанский*, *венский* и *офисный* пронумерованы подряд от 1 до 6, а детали пронумеровали подряд от 1 до 11. В строке 7 приводится задание множества стульев I, в строке 8 — задание множества деталей J. Между именем множества и перечислением элементов, входящих в это множество, находится поясняющий текст.

В именах множеств, таблиц, векторов, переменных, ограничений и пр. нельзя использовать кириллицу. В конце каждого блока ставится знак «;», строка 9.

Строки 10—29 соответствуют описанию векторов p_i и q_j. Между двумя наклонными чертами «$/\ldots/$» заключены пары: координата вектора и значение по этой координате. В конце блока ставится «;».

Строки 30—43 соответствуют описанию матрицы (d_{ji}). Обратите внимание, что каждый столбец в таблице выровнен по левому (допускается выравнивание по правому) краю.

В строках 44—48 декларируются переменные. В задаче имеется шесть целочисленных неотрицательных переменных $x(i)$. На языке GAMS они объявляются Integer Variables. По умолчанию диапазон значений переменных типа Integer от 0 до 100. В данном примере это ограничение может оказаться существенным, поэтому необходимо расширить диапазон Integer. Для этого в строке 46 изменяется максимальное значение переменных $x(i)$ с помощью команды $x.up(i) = INF$, где INF — это некоторое большое число в GAMS, аналог бесконечности. Переменная z, которая соответствует значению целевой функции, должна быть свободной. На языке GAMS она объявляется Free Variable («Free» можно опустить).

Строки 49—53 соответствуют блоку Equations. В нем объявляются имена всех ограничений, чтобы в дальнейшем на них ссылаться, и выписываются ограничения. Символы две точки подряд «..» всегда разделяют название ограничения и начало алгебраического выражения. Символы $= e =$ означают равенство (от англ. equal — равно); $= l =$ означают меньше либо равно (от англ. less — меньше); $= g =$ означают больше, либо равно (от англ. greater — больше).

После того, как все ограничения описаны, необходимо указать, какие из них нужно учитывать при решении задачи. Возможно, что какие-то ограничения оказались несущественными, или Вы хотите решить задачу несколько раз с разными наборами ограничений.

В строке 54 с помощью оператора $Model$ присваивается название модели и перечисляются именно те ограничения, которые следует учесть при решении задачи. В данном примере модель называется $Chairs$. Между двумя наклонными чертами $/\ldots/$ перечислены те ограничения, которые вошли в модель.

Поскольку все описанные ограничения формируют модель, то можно записать это в более компактной форме с помощью ключевого слова *all* (от англ. all — все), строка 55.

Строки 56, 57 соответствуют подготовке и обращению к решателю. Для того, чтобы вызвать решатель, нужно в операторе *Solve* указать название решаемой модели, её тип и критерий оптимизации. Модель *Chairs* относится к целочисленному линейному программированию. Критерий оптимизации — максимизация дохода z.

Для задач минимизации используется ключевое слово min. Кроме задач класса *mip*, можно решать и другие. Это зависит от возможностей установленного решателя. В меню *File* --→ *Options* --→ *Solvers* можно подключить соответствующий решатель.

Перед тем, как вызывать решатель, можно сделать ряд настроек, улучшающих качество получаемого решения. Так, в строке 56 устанавливается относительный разрыв (*relativegap*), равный 0 %, который по умолчанию составляет 10 %.

В результате выполнения оператора *Solve*:
— подготовлены данные в соответствующей структуре, чтобы запустить решатель;
— запущен решатель;
— получено решение или установлена неразрешимость задачи.

Результаты расчетов выводятся в отдельный файл с расширением .lst. Получить оптимальное решение можно с помощью оператора *Display*, как это сделано в строке 58. В результате выполнения этого оператора будут напечатаны значения переменных x и z в файле с расширением .lst после ключевого слова Execution.

Для запуска программы выберите меню *File* --→ *Run*. Появится вспомогательное окно, в котором решатель сообщит о результатах компиляции и поиска решений. В случае успешного завершения работы решателя будет создан новый файл с расширением .lst, в котором появится полная информация о модели, процессе вычислений, статистике модели и результатах работы алгоритма.

Ниже приводятся результаты расчетов для рассмотренного примера.

```
**** REPORT SUMMARY :        0     NONOPT
                             0 INFEASIBLE
                             0  UNBOUNDED
G e n e r a l   A l g e b r a i c   M o d e l i n g   S y s t e m
E x e c u t i o n

58 VARIABLE x.L  сколько стульев $i$-го типа производит фирма

1 100.000,    2  72.000,    3  40.000,    4  53.000

58 VARIABLE z.L = 10294.000  суммарный доход от продажи стульев
```

```
58 EQUATION constraints1.L  ограничения на кол-во деталей на складе

1  1280.000,  2  1900.000,  3  1060.000,  4   100.000,  5   165.000
6   760.000,  7   553.000,  8   100.000,  9    72.000,  10   93.000

EXECUTION TIME   =    0.000 SECONDS
```

Суммарный доход составляет 10294. Оптимальный план производства состоит из 100 стульев типа *капитан*, 72 стульев типа *помощник*, 40 стульев типа *маркиза* и 53 *испанских* стульев, остальные типы стульев в оптимальном решении отсутствуют. Дополнительно приводится информация о том, сколько деталей было задействовано при реализации оптимального плана. Эта информация может оказаться полезной при анализе решения.

Кроме упомянутых выше возможностей, в GAMS заложены и другие средства для реализации сложных задач и взаимодействия с приложениями [18].

7. Список литературы

1. Алексеева Е. В., Кочетов Ю. А. Курс лекций по теории принятия решений. URL: `http://www.math.nsc.ru/LBRT/k5/or.html`

2. Вентцель Е. С. Исследование операций. Задачи, принципы, методология. М.: КноРус, 2010.

3. Вершик А. М. Леонид Витальевич Канторович: человек и ученый: В 2 т. Новосибирск: Изд-во СО РАН. Филиал «Гео», 2002. Т. 1.

4. Гольштейн Е. Г., Юдин Д. Б. Задачи линейного программирования транспортного типа. М.: Либроком, 2010.

5. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.

6. Емеличев В. А., Ковалев М. М., Кравцов М. К. Многогранники графы, оптимизация (комбинаторная теория многогранников). М.: Наука, 1981.

7. Есипов Б. А. Методы исследования операций: Учебное пособие. СПб.: Лань, 2010.

8. Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К. Алгоритмы: построение и анализ. М.: Вильямс, 2009.

9. Кочетов Ю. А. Методы локального поиска для дискретных задач размещения Модели и алгоритмы. Saarbrucken: Lambert Academic Publishing, 2011.

10. Кочетов Ю. А., Плясунов А. В. Генетический локальный поиск для задачи о разбиении графа на доли ограниченной мощности // Журнал вычислительной математики и математической физики. 2012. Т. 52. № 1.

11. Ларичев О. И. Теория и методы принятия решений, а также Хроника событий в Волшебных Странах: Учебник. М.: Логос, 2000.

12. Лэсдон Л. Оптимизация больших систем. М.: Наука, 1975.

13. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. М.: Мир, 1985.

14. Ларин Р. М., Плясунов А. В., Пяткин А. В. Методы оптимизации. Примеры и задачи: Учеб. пособие, 2-е изд. Новосибирск, 2009.

15. Программное обеспечение для решения оптимизационных задач Advanced Integrated Mathematical Modeling System AIMMS. URL: `www.aimms.com`

16. Программное обеспечение для решения оптимизационных задач AMPL Optimization. URL: `www.ampl.com`

17. Программное обеспечение для решения оптимизационных задач IBM ILOG. URL: `www.ibm.com/software/websphere/products/optimization`

18. Программное обеспечение для решения оптимизационных задач GAMS. URL: `http://gams.com/`

19. Программное обеспечение для решения оптимизационных задач GNU Linear Programming Kit Package. URL: `http://www.gnu.org/software/glpk/glpk.html`

20. Программное обеспечение для решения оптимизационных задач Microsoft Office, надстройка «Поиск решения» в Excel. URL: `http://www.solver.com`

21. Сигал И. Х., Иванова А. П. Введение в прикладное дискретное программ-рование программирование: модели и вычислительные алгоритмы: Учеб. пособие. М.: ФИЗМАТЛИТ, 2007.

22. Схрайвер А. Теория линейного и целочисленного программирования: В 2 т. М.: Мир, 1991.

23. Appa G. M., Pitsoulis L. S., Paul W. H. (Eds.) Handbook on Modelling for Discrete Optimization. Springer Series Vol. International Series in Operations Research & Management Science 2006. V. 88, XXII.

24. Eppen G. D., Gould F. J., Schmidt C. P. Introductory management science. 4^{th} Edition. A Simon & Schuster Company. 1993.

25. Pochet Y., Wolsey L. A. Production planning by mixed integer programming. In: T. V. Mikosh, S. I. Resnick, S. M. Robinson (Eds.) Springer Series in Operations Research & Financial Engineering. 2006.

26. Plastria F. Formulating logical implications in combinatorial optimization // European journal of Operational Research. 2002. № 140.

27. Ehrgott M. Multicriteria Optimization. 2^{nd} Edition. Berlin: Springer, 2005.

28. Eiselt H. A., Sandblom C.–L. Operations Research: A Model-Based Approach. Berlin, Heidelberg: Springer-Verlag, 2010.

29. Kellerer H., Pferschy U., Pisinger D. Knapsack problems. Berlin, Heidelberg: Springer-Verlag, 2004.

30. Nemhauser G. L., Wolsey L. A. Integer and combinatorial optimization. John Wiley & Sons, Inc., 1999.

31. Williams H. P. Model Building in Mathematical Programming. 4^{th} Edition, University of Southampton, John Wiley & Sons, Inc., 1999.

Учебное издание

Алексеева Екатерина Вячеславовна

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ЦЕЛОЧИСЛЕННОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ. ПРИМЕРЫ И ЗАДАЧИ

Учебное пособие

Редактор *Е. П. Войтенко*

Подписано в печать 26.03.2012г.
Формат 60×84 1/16. Печать офсетная.
Уч.-изд.л. 8. Усл. печ. л. 7,44. Тираж 200 экз.

Заказ №

Редакционно-издательский центр НГУ.
630090, Новосибирск - 90, ул. Пирогова, 2.