

Линейное и целочисленное линейное программирование

учебное пособие ¹

В.Н.Шевченко, Н.Ю.Золотых

2002

¹Нижегородский государственный университет

Оглавление

Обозначения	4
1. Введение	5
1.1. Задача математического программирования	5
1.2. Задача выпуклого программирования	6
1.3. Задача линейного программирования	12
1.4. Основная идея симплекс-метода	15
1.5. Примеры задач линейного программирования	16
1.5.1. Задача максимизации прибыли	16
1.5.2. Задача о «смесях»	17
1.5.3. Транспортная задача	17
1.5.4. Задачи о назначениях	18
1.5.5. Задача о «раскрое»	19
1.5.6. Задача коммивояжера	20
1.6. Задачи	22
2. Симплекс-метод	23
2.1. Числовой пример	23
2.2. Симплекс-метод в строчной форме	26
2.3. Зацикливание и способы защиты от него	32
2.3.1. Зацикливание	32
2.3.2. Лексикографический метод	36
2.3.3. Правило Бленда выбора ведущего элемента	38
2.4. Получение начального допустимого опорного плана	39
2.5. Задачи	43
2.6. Столбцовая форма	44

<i>ОГЛАВЛЕНИЕ</i>	3
3. Двойственность в линейном программировании	50
3.1. Теорема двойственности	50
3.2. Дополняющая нежесткость в линейном программировании	57
3.3. Задачи	59
3.4. Двойственный симплекс-метод	60
3.5. Задачи	62
4. Целочисленное линейное программирование	63
4.1. Идея правильных отсечений	63
4.2. Постановка задачи	64
4.3. Циклический алгоритм Гомори	64
4.4. Полностью целочисленный алгоритм	68
4.4.1. Прямой метод целочисленного программирования	71
4.5. Задачи	73
Программа курса	74
Литература	76

Обозначения

\mathbf{Z} кольцо целых чисел,

\mathbf{Q} поле рациональных чисел,

\mathbf{R} поле вещественных чисел,

P^n множество n -мерных арифметических векторов с компонентами из P (рассматриваемых как строки или как столбцы),

$P^{n \times m}$ множество матриц размером $n \times m$ с элементами из P ,

$[\alpha]$ целая часть числа α : наибольшее целое, не превосходящее α .

$\{\alpha\}$ дробная часть числа α : $\{\alpha\} = \alpha - [\alpha]$.

Глава 1

Введение

1.1. Задача математического программирования

До XIX века основным поставщиком прикладных задач для математики были астрономия, механика, физика, а основной и весьма плодотворной идеей — идея непрерывности, приведшая к становлению мощного аппарата интегрально-дифференциального исчисления. Развитие экономики привело к возможности рассмотрения количественных закономерностей и в рамках этой науки; с появлением экономических моделей Кенэ (1758 г.), Маркса, Вальраса и др. по существу началась математическая экономика.

В 1939 году вышла в свет монография Л.В. Канторовича¹ «Математические методы организации и планирования производства», где выявлен широкий класс производственно-экономических оптимизационных задач, допускающих строгое математическое описание. Идеи, содержащиеся в этой книге, были затем им развиты и привели к созданию линейного программирования. Начиная с 1942 года, исследования по линейному, а позже и нелинейному программированию активно ведутся и в США.

Таким образом, в середине XX века выкристаллизовались оптимизационные задачи двух типов: максимизации прибыли при ограниченных

¹Канторович Леонид Витальевич (1912–1986) — советский математик и экономист. Академик АН СССР (1964, чл. корр. 1958). Гос. премия СССР 1949, Ленинская премия (1965), Нобелевская премия (1975, совместно с американским экономистом Т. И. Купмансом).

ресурсах и минимизации расходов для осуществления заданного комплекса работ. Рассмотрим подробнее первую из них. Пусть план работы предприятия можно охарактеризовать вектором x , по x однозначно находится вектор $y = F(x)$, описывающий затраты ресурсов, и число $f(x)$, соответствующее прибыли. Возможно, что на x нужно наложить еще ограничения технологического характера, которые могут принимать вид равенств, неравенств или описываться более сложно. Вектор x , при котором $F(x)$ удовлетворяет ограничениям на ресурсы и технологию, назовем *планом* или *допустимым вектором*, и множество планов обозначим через M . Назовем план x_0 *оптимальным*, если $f(x_0) \geq f(x)$ для всякого x из M . Заметим, что оптимальный план может и не существовать, даже если $M \neq \emptyset$.

Будем понимать под *задачей математического программирования* решение следующих вопросов:

- 1) существует ли $x \in M$;
- 2) существует ли оптимальный план x_0 ;
- 3) найти какой-нибудь оптимальный план x_0 и вычислить $f(x_0)$.

Для сокращенного обозначения задачи математического программирования введем запись

$$\min_{x \in M} f(x) \quad (1)$$

Поскольку, с одной стороны, общего алгоритма, решающего поставленную задачу даже при некоторых частных способах задания множества M и функции $f(x)$, не существует, а, с другой стороны, имеется много важных приложений, приводящих к классам задач вида (1), в которых M и $f(x)$ удовлетворяет каким-то дополнительным предположениям, то рассмотрим некоторые из этих классов.

1.2. Задача выпуклого программирования

Определение 1.1. Множество M называется *выпуклым*, если вместе с любыми двумя своими точками y и z оно содержит и все точки вида $x(\alpha) = (1 - \alpha)y + \alpha z$, где $0 \leq \alpha \leq 1$.

Так как множество точек вида $x(\alpha) = (1 - \alpha)y + \alpha z$, где $0 \leq \alpha \leq 1$, представляет собой отрезок $[y, z]$, то определение выпуклого множества может быть переформулировано следующим образом.

Определение 1.2. Множество M называется *выпуклым*, если вместе с любыми двумя своими точками z и y оно содержит весь отрезок, соединяющий эти точки.

Приведем примеры выпуклых множеств.

Пример 1.3. Замкнутое полупространство — это множество

$$H(a, \beta) = \left\{ x : \sum_{i=1}^n a_i x_i \leq \beta \right\},$$

где $a = (a_1, a_2, \dots, a_n)$, $a \neq 0$.

Покажем, что замкнутое полупространство — выпукло. Пусть $y = (y_1, y_2, \dots, y_n)$ и $z = (z_1, z_2, \dots, z_n)$ две точки замкнутого полупространства $H(a, \beta)$, α — произвольное число, $0 \leq \alpha \leq 1$. Надо доказать, что точка $x(\alpha) = \alpha z + (1 - \alpha)y$ также принадлежит $H(a, \beta)$, то есть убедиться в справедливости неравенства

$$\sum_{i=1}^n a_i (\alpha z_i + (1 - \alpha)y_i) \leq \beta.$$

По определению для точек y и z выполняются неравенства

$$\sum_{i=1}^n a_i z_i \leq \beta \quad \text{и} \quad \sum_{i=1}^n a_i y_i \leq \beta.$$

Умножая эти неравенства соответственно на неотрицательные числа α и $(1 - \alpha)$, а затем складывая, получим требуемое неравенство.

Пример 1.4. Открытое полупространство:

$$H^\sim(a, \beta) = \left\{ x : \sum_{i=1}^n a_i x_i < \beta \right\},$$

где $a = (a_1, a_2, \dots, a_n)$, $a \neq 0$. — также является выпуклым множеством.

Пример 1.5. Шар размерности n радиуса r :

$$B_n(r) = \left\{ x : \sum_{i=1}^n x_i^2 \leq r^2 \right\}.$$

Убедимся, что $B_n(r)$ — выпуклое множество. Пусть $y = (y_1, y_2, \dots, y_n)$ и $z = (z_1, z_2, \dots, z_n)$ — точки $B_n(r)$, α — произвольное число, $0 \leq \alpha \leq 1$. Покажем, что точка

$$x(\alpha) = (\alpha z_1 + (1 - \alpha)y_1, \dots, \alpha z_n + (1 - \alpha)y_n)$$

принадлежит $B_n(r)$. Проведем преобразования:

$$\sum_{i=1}^n (\alpha z_i + (1 - \alpha)y_i)^2 = \alpha^2 \sum_{i=1}^n z_i^2 + 2\alpha(1 - \alpha) \sum_{i=1}^n y_i z_i + (1 - \alpha)^2 \sum_{i=1}^n y_i^2.$$

Согласно неравенству Коши–Буняковского имеем:

$$\sum_{i=1}^n y_i z_i \leq \sqrt{\sum_{i=1}^n y_i^2} \cdot \sqrt{\sum_{i=1}^n z_i^2}.$$

Заменяя во всех выражениях

$$\sum_{i=1}^n y_i^2 \quad \text{и} \quad \sum_{i=1}^n z_i^2$$

на r^2 , получаем неравенство

$$\sum_{i=1}^n (\alpha z_i + (1 - \alpha)y_i)^2 \leq (\alpha^2 + 2\alpha(1 - \alpha) + (1 - \alpha)^2)r^2 = r^2,$$

что и требовалось.

Определение 1.6. Точка $\alpha_1 x_1 + \dots + \alpha_s x_s$, где $\alpha_1 + \dots + \alpha_s = 1$, называется *аффинной комбинацией* точек x_1, \dots, x_s . Множество всех аффинных комбинаций точек из M называется *аффинной оболочкой* множества M и обозначается $\text{Aff } M$.

С выпуклым множеством тесно связано понятие выпуклого конуса.

Определение 1.7. Множество M называется *выпуклым конусом*, если вместе с любыми двумя своими точками y и z оно содержит и все точки вида $\alpha y + \beta z$, где $\alpha \geq 0$, $\beta \geq 0$.

Определение 1.8. Точка $\alpha_1 x_1 + \dots + \alpha_s x_s$, где $\alpha_1 \geq 0, \dots, \alpha_s \geq 0$, называется *неотрицательной комбинацией* точек x_1, \dots, x_s . Множество всех неотрицательных комбинаций точек из M называется *конической оболочкой* множества M и обозначается $\text{Cone } M$.

Определение 1.9. Точка $\alpha_1 x_1 + \dots + \alpha_s x_s$, где $\alpha_1 \geq 0, \dots, \alpha_s \geq 0$, $\alpha_1 + \dots + \alpha_s = 1$, называется *выпуклой комбинацией* точек x_1, \dots, x_s . Множество всех выпуклых комбинаций точек из M называется *выпуклой оболочкой* множества M и обозначается $\text{Conv } M$.

Нетрудно проверить выпуклость множеств $\text{Aff } M$, $\text{Cone } M$, $\text{Conv } M$ (см. упражнения 1.2–1.5).

Определение 1.10. Если M — конечное множество, то $\text{Conv } M$ называется *выпуклым политопом*, или *выпуклым многогранником*.

Лемма 1.11. *Пересечение выпуклых множеств выпукло.*

ДОКАЗАТЕЛЬСТВО. Пусть $\{M_i, i \in I\}$ — система (возможно бесконечная) выпуклых множеств и

$$M = \bigcap_{i \in I} M_i.$$

Пусть y и z точки из M , $0 \leq \alpha \leq 1$. Поскольку для каждого i справедливо включение $M \subseteq M_i$, то точки y и z принадлежат M_i . Так как M_i — выпуклое множество, то точка $(1 - \alpha)y + \alpha z$ принадлежит M_i , а, значит, и пересечению $M = \bigcap M_i$. ■

Определение 1.12. *Гиперплоскость* — это множество точек вида:

$$G(a, \beta) = \left\{ x : \sum_{i=1}^n a_i x_i = \beta \right\}.$$

Выпуклость $G(a, \beta)$ следует из равенства $G(a, \beta) = H(a, \beta) \cap H(-a, -\beta)$ и леммы 1.11.

Определение 1.13. *Полиэдром*, или *многогранным множеством* называется множество решений системы линейных уравнений и нестрогих неравенств.

Выпуклость полиэдра также легко следует из леммы 1.11.

Определение 1.14. Функция $f(x)$ называется *выпуклой* на выпуклом множестве M , если для любых двух точек x и y из M и любого числа α такого, что $0 \leq \alpha \leq 1$, выполняется неравенство

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y).$$

Индукцией по s легко доказывается следующая лемма.

Лемма 1.15. Если $f(x)$ выпукла на M , $x_i \in M$, $\alpha_i \geq 0$ ($i = 1, \dots, s$) и $\sum_{i=1}^s \alpha_i = 1$, то

$$f\left(\sum_{i=1}^s \alpha_i x_i\right) \leq \sum_{i=1}^s \alpha_i f(x_i).$$

Заметим, что если функция $f(x)$ — выпуклая на выпуклом множестве M , то множество $\{x \in M : f(x) \leq \beta\}$ также выпукло. В частности выпуклость (на всем пространстве) множества $H(\alpha, \beta)$ следует из выпуклости линейной функции

$$f(x) = \sum_{i=1}^n \alpha_i x_i,$$

а выпуклость шара $B_n(r)$ вытекает из выпуклости функции

$$f(x) = \sum_{i=1}^n x_i^2.$$

Определение 1.16. Задача минимизации выпуклой функции на выпуклом множестве называется *задачей выпуклого программирования*.

Задача выпуклого программирования является частным случаем задачи математического программирования (1), если M выпуклое множество, а функция $f(x)$ выпукла на M .

Важным свойством задачи выпуклого программирования является совпадение локального и глобального минимума.

Определение 1.17. Точка x_0 из M называется *локальным минимумом* функции $f(x)$ на M , если существует такое $\varepsilon > 0$, что для любого y из M , удовлетворяющего неравенству $|y - x_0| < \varepsilon$ выполняется неравенство $f(x_0) \leq f(y)$.

Другими словами, точка x_0 является точкой локального минимума, если найдется такая ε -окрестность точки x_0 , что значение функции $f(x)$ в любой точке этой окрестности не меньше $f(x_0)$.

Теорема 1.18. *Всякий локальный минимум в задаче выпуклого программирования является оптимальным планом.*

ДОКАЗАТЕЛЬСТВО. Пусть x_0 — локальный минимум выпуклой на M функции $f(x)$ и вопреки доказываемому утверждению существует $x_1 \in M$ такой, что $f(x_1) < f(x_0)$. По определению выпуклой функции, при $0 < \alpha \leq 1$ выполняются неравенства

$$f(\alpha x_1 + (1 - \alpha)x_0) \leq \alpha f(x_1) + (1 - \alpha)f(x_0) < f(x_0).$$

Однако это противоречит локальной минимальности x_0 . Действительно, для любого $\varepsilon > 0$ точка $z = \alpha x_1 + (1 - \alpha)x_0$, где

$$\alpha = \min \left\{ \frac{1}{2}, \frac{\varepsilon}{2|x_1 - x_0|} \right\},$$

принадлежит как M (в силу выпуклости), так и ε -окрестности точки x_0 (так как $|z - x_0| = \alpha|x_1 - x_0| < \varepsilon$). ■

Таким образом, чтобы определить, достигает ли выпуклая на выпуклом множестве M функция $f(x)$ минимума в точке x_0 , достаточно рассмотреть поведение $f(x)$ вблизи x_0 ; если x_0 неоптимальный, то найдется отрезок (направление), двигаясь по которому можно получить новый план с меньшим, чем в x_0 , значением $f(x)$. Методы, основанные на этой идее, называются *градиентными*.

Не так обстоит дело, если речь идет о максимизации выпуклой на выпуклом множестве M функции $f(x)$. В таких задачах для нахождения оптимального плана, вообще говоря (как правило, так оно и бывает), необходимо перебрать все локальные максимумы (см. упражнения 1.13–1.14). Однако, если в рассматриваемом случае M задано в виде $M = \text{Conv} \{x_1, \dots, x_s\}$, то имеет место следующая теорема.

Теорема 1.19. *Если $M = \text{Conv} \{x_1, \dots, x_s\}$ и $f(x)$ — выпуклая на M функция, то среди планов x_1, \dots, x_s есть оптимальный план задачи (1).*

ДОКАЗАТЕЛЬСТВО. Пусть k такое число, что

$$f(x_k) = \max_{i=1, \dots, s} f(x_i).$$

Тогда для любого $x \in \text{Conv} \{x_1, \dots, x_s\}$ имеет место

$$f(x) = f\left(\sum_i \alpha_i x_i\right) \leq \sum_i \alpha_i f(x_i) \leq f(x_k),$$

т.е. x_k — оптимальный план задачи (1). ■

Разумеется, Теорема 1.19 сводит дело к конечному перебору, однако число s в практически важных задачах настолько велико, что осуществить этот перебор не под силу и компьютеру. Кроме того, известные из математического анализа условия, выделяют точки, «подозрительные» на локальный экстремум, лишь внутри области и, следовательно, здесь не работают.

Пусть теперь $M = \text{Conv} \{x_1, \dots, x_s\}$, а $f(x)$ — линейная функция. В этом случае применимы и Теорема 1.18 и Теорема 1.19, и полный перебор планов x_1, \dots, x_s можно попытаться заменить их упорядоченным перебором. Симплекс-метод, рассматриваемый ниже, является конкретизацией этой идеи.

1.3. Задача линейного программирования

Определение 1.20. Задача минимизации или максимизации линейной функции (называемой *целевой*) на полиэдре M называется *задачей линейного программирования*, сокращенно ЗЛП.

Согласно определению, задача линейного программирования может быть записана в виде

$$\begin{cases} \min(c_1 x_1 + c_2 x_2 + \dots + c_n x_n) \\ a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n \leq b_i & (i = 1, 2, \dots, k), \\ a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n \geq b_i & (i = k + 1, \dots, k + s), \\ a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n = b_i & (i = k + s + 1, \dots, m), \end{cases}$$

где величины c_j , a_{ij} , b_i ($j = 1, 2, \dots, n$, $i = 1, 2, \dots, m$) являются элементами некоторого подполя F поля вещественных чисел \mathbf{R} . Умножив

каждое неравенство во второй группе на -1 и переименовав коэффициенты, представим ЗЛП в виде:

$$\min(c_1x_1 + c_2x_2 + \dots + c_nx_n) \quad (2)$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i \quad (i = 1, 2, \dots, s), \quad (3)$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i \quad (i = s + 1, \dots, m). \quad (4)$$

ЗЛП (2-4) называется *общей*. Кроме общей формы ЗЛП выделяют также каноническую и стандартную формы. ЗЛП вида

$$\min(c_1x_1 + c_2x_2 + \dots + c_nx_n) \quad (5)$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i \quad (i = 1, 2, \dots, m), \quad (6)$$

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0 \quad (7)$$

называется *канонической*, а ЗЛП вида

$$\max(c_1x_1 + c_2x_2 + \dots + c_nx_n) \quad (8)$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i \quad (i = 1, 2, \dots, m), \quad (9)$$

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0 \quad (10)$$

называется *стандартной*.

Пусть A — матрица $m \times n$ с элементами a_{ij} , b — вектор-столбец $(b_1, \dots, b_m)^T$, c — вектор-строка (c_1, \dots, c_n) . Если $a = (a_1, \dots, a_n)$ и $b = (b_1, \dots, b_n)$, то под неравенством $a \geq b$ будем понимать n неравенств $a_i \geq b_i$ ($i = 1, \dots, n$).

Используя матричные операции, запишем каноническую ЗЛП и стандартную ЗЛП соответственно в виде

$$\begin{array}{l} \min cx \\ \left\{ \begin{array}{l} Ax = b, \\ x \geq 0 \end{array} \right. \end{array} \quad \text{и} \quad \begin{array}{l} \max cx \\ \left\{ \begin{array}{l} Ax \leq b, \\ x \geq 0. \end{array} \right. \end{array}$$

Для аналогичного представления общей ЗЛП понадобится блочное представление:

$$A = \left(\begin{array}{c} A' \\ A'' \end{array} \right), \quad \left(\begin{array}{c} b' \\ b'' \end{array} \right),$$

где A' и b' состоят из первых s строк матрицы A и первых s компонент вектора b , а A'' и b'' — из остальных строчек матрицы A и компонент вектора b , соответственно. Тогда общая задача линейного программирования представляется в виде

$$\begin{aligned} & \min cx \\ & \begin{cases} A'x \leq b', \\ A''x = b''. \end{cases} \end{aligned}$$

Все формы ЗЛП эквивалентны в том смысле, что каждую из них можно преобразовать к любой другой из форм. Покажем, как привести общую ЗЛП к стандартной. Для этого каждое уравнение $a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i$ из (6) заменим двумя неравенствами

$$\begin{aligned} a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n &\leq b_i, \\ -a_{i1}x_1 - a_{i2}x_2 - \dots - a_{in}x_n &\leq -b_i, \end{aligned}$$

положим $x = u - v$ и потребуем выполнение неравенств $u \geq 0, v \geq 0$. Общая задача линейного программирования в новых переменных будет иметь вид стандартной ЗЛП:

$$\begin{aligned} & \max(cv - cu) \\ & \begin{cases} A'u - A'v \leq b', \\ A''u - A''v \leq b'', \\ -A''u + A''v \leq -b'', \\ u \geq 0, v \geq 0. \end{cases} \end{aligned} \quad (11)$$

Если общая задача линейного программирования имеет решение x , то стандартная ЗЛП (11) тоже имеет решение, которое можно найти по формулам:

$$u_i = \begin{cases} x_i, & \text{если } x_i \geq 0, \\ 0, & \text{если } x_i < 0 \end{cases} \quad \text{и} \quad v_i = \begin{cases} 0, & \text{если } x_i \geq 0, \\ -x_i, & \text{если } x_i < 0. \end{cases}$$

С другой стороны по решению (u, v) стандартной ЗЛП (11) можно найти решение $x = u - v$ общей ЗЛП. Очевидно, что значения целевых функций на соответствующих планах равны по абсолютной величине и противоположны по знаку. Таким образом, общая задача линейного программирования (2–4) легко сводится к стандартной задаче линейного программирования (11).

Приведем стандартную ЗЛП (8–10) к канонической ЗЛП. Для этого введем вектор y , число компонент которого совпадает с числом неравенств в (9). Рассмотрим каноническую ЗЛП

$$\begin{cases} \min(-cx) \\ Ax + y = b, \\ x \geq 0, \\ y \geq 0. \end{cases} \quad (12)$$

Легко видеть, что между решениями x стандартной задачи линейного программирования (8–10) и решениями (x, y) канонической задачи линейного программирования (12) существует взаимно однозначное соответствие $x \leftrightarrow (x, b - Ax)$. Таким образом, стандартная задача линейного программирования сводится к канонической задаче.

Так как каноническая ЗЛП является частным случаем общей задачи линейного программирования, то эквивалентность всех форм ЗЛП установлена.

1.4. Основная идея симплекс-метода

Попробуем показать, как из теорем 1.18 и 1.19 можно получить алгоритм решения ЗЛП. Более формальное изложение этого алгоритма, носящего название *симплекс-метод*, находится в последующих разделах.

Рассмотрим стандартную ЗЛП (8–10). Если $b \geq 0$, то $x = 0$ — допустимый план. Если при этом $c \leq 0$, то для любого допустимого плана $cx \leq 0$, следовательно, нулевой вектор является оптимальным планом. Если найдется такое j , при котором $c_j > 0$, то попытаемся увеличить x_j , не меняя при этом других координат и сохраняя условие допустимости плана (ср. с теоремой 1.18). Ясно, что если для всех $i = 1, \dots, m$ выполнено неравенство $a_{ij} \leq 0$, то величину x_j можно увеличивать до бесконечности и, следовательно, целевая функция на множестве допустимых планов не ограничена сверху. Если же найдется такое i , при котором $a_{ij} > 0$, то неравенство

$$\sum_{k=1}^n a_{ik}x_k \leq b_i$$

ограничивает рост x_j величиной b_i/a_{ij} . Положим

$$\alpha = \min_{i:a_{ij}>0} \frac{b_i}{a_{ij}} = \frac{b_k}{a_{kj}}.$$

Если $\alpha > 0$, то получим новый допустимый вектор αe_j , у которого все компоненты равны 0, кроме j -й координаты, равной α . Значение линейной формы на нем равно $\alpha c_j > 0$. Можно проверить (формально это будет вытекать из результатов раздела 2.2), что замена переменной x_j на переменную

$$y_k = b_i - \sum_{j=1}^n a_{ij} x_j$$

приведет к стандартной задаче с неотрицательной правой частью и процесс можно повторить. Указанную замену переменных можно выполнить и при $\alpha = 0$ (так и делается), однако при этом «новый» план равен «старому».

Заметим, что существуют примеры, показывающие, что условие $c \leq 0$ не является необходимым условием оптимальности точки $x = 0$ (см. задачу 1.14).

1.5. Примеры задач линейного программирования

1.5.1. Задача максимизации прибыли

Пусть имеется m видов ресурсов в количестве b_1, b_2, \dots, b_m , которые могут быть использованы при производстве n видов изделий. Известны числа a_{ij} ($i = 1, \dots, m; j = 1, \dots, n$), характеризующие количество i -го ресурса, необходимого для производства j -го изделия. Кроме этого, известна стоимость c_j единицы j -го изделия. Необходимо найти план выпуска изделий, максимизирующий прибыль от продажи всех изделий.

Обозначив через x_j количество j -го изделия, получим математическую модель:

$$\begin{cases} \max \sum_{j=1}^n c_j x_j \\ \sum_{j=1}^n a_{ij} x_j \leq b_i & (i = 1, \dots, m), \\ x_j \geq 0 & (j = 1, \dots, n). \end{cases}$$

Заметим, что в данной задаче на некоторые переменные может быть наложено требование целочисленности.

1.5.2. Задача о «смесях»

Имеется n компонентов, при сочетании которых в различных пропорциях образуются различные смеси. В состав каждого компонента входят m веществ. Пусть a_{ij} означает количество i -го вещества ($i = 1, \dots, m$), которое входит в состав единицы j -го компонента ($j = 1, \dots, n$), а a_i — количество i -го вещества, которое входит в единицу смеси. Заданы числа c_j ($j = 1, \dots, n$), характеризующие цену единицы j -го компонента, и числа b_i ($i = 1, \dots, m$), указывающие минимально необходимое содержание i -го вещества в смеси. Требуется определить состав смеси минимальной стоимости.

Пусть числа x_j ($j = 1, \dots, n$) характеризуют количество j -ой компоненты в смеси. Тогда задача принимает следующий вид. Требуется минимизировать функцию, характеризующую общую стоимость:

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

при условиях обязательного минимального содержания каждого вещества:

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

По-видимому, одной из первых практических задач такого рода была задача составления диеты, поставленная и решенная в 1947 г. в США в Национальном бюро стандартов. Рассматриваемая система имела параметры $m = 9$, $n = 77$ (см. [?], с. 521–527).

1.5.3. Транспортная задача

Пусть на m складах имеются различные количества некоторого товара, который необходимо доставить в n пунктов назначения. Точнее, i -ый склад располагает количеством товара, равным a_i , тогда как j -ый пункт назначения должен получить b_j единиц товара. Предполагается, что суммарный спрос равен суммарному предложению, т.е.

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

Положим

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-ое лицо назначается на } j\text{-ую работу,} \\ 0 & \text{в противном случае.} \end{cases} \quad (13)$$

Так как каждое лицо может быть назначено только на одну работу, то

$$\sum_{j=1}^n x_{ij} = 1 \quad (i = 1, \dots, n). \quad (14)$$

Так как каждая работа предназначается только для одного лица, то

$$\sum_{i=1}^n x_{ij} = 1 \quad (j = 1, \dots, n). \quad (15)$$

Суммарная полезность выражается формулой:

$$\sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}. \quad (16)$$

Итак, в математической постановке задача о назначениях заключается в максимизации функции (16), при ограничениях (13–15).

В данной постановке задача о назначениях не является задачей линейного программирования. Этому препятствуют условия (13). Однако можно показать, что в данном случае условия (13) можно заменить условиями

$$x_{ij} \geq 0 \quad (i = 1, \dots, n; j = 1, \dots, n),$$

сведя задачу о назначениях к частному случаю транспортной задачи.

1.5.5. Задача о «раскрое»

Из некоторого материала требуется изготовить m различных изделий в количестве, пропорциональном числам b_1, \dots, b_m . Каждая единица материала может быть раскроена n различными способами, причем использование j -го способа дает a_{ij} единиц i -го изделия. Найти план раскроя a единиц материала, обеспечивающий максимальное число комплектов изделий.

Обозначим через x_j количество единиц материала, раскраиваемого j -ым способом. Переменные x_j должны удовлетворять, очевидно, следующим ограничениям:

$$\begin{cases} x_j \geq 0, \\ x_j \in \mathbf{Z}, \\ \sum_{j=1}^n x_j = a, \\ \sum_{j=1}^n a_{ij} x_j = b_i x \quad (i = 1, \dots, m), \end{cases} \quad (17)$$

где x — число комплектов изделий.

Задача заключается в максимизации x при ограничениях (17).

1.5.6. Задача коммивояжера

Коммивояжер должен посетить n городов, не побывав ни в одном дважды и в конце возвратиться в исходный город. Предположим, что известны все расстояния a_{ij} между любыми двумя городами i и j ($i = 1, \dots, n; j = 1, \dots, n$). Каков должен быть маршрут коммивояжера, чтобы суммарное расстояние, пройденное им, было минимальным?

Известно несколько математических постановок этой задачи. Рассмотрим лишь одну из них (Таккер).

Определим переменные x_{ij} следующим образом:

$$x_{ij} = \begin{cases} 1, & \text{если коммивояжер переезжает из города } i \text{ в город } j, \\ 0, & \text{в противном случае.} \end{cases} \quad (18)$$

Из каждого города коммивояжер должен выехать ровно один раз, поэтому

$$\sum_{j=1}^n x_{ij} = 1 \quad (i = 1, \dots, n). \quad (19)$$

В каждый город коммивояжер должен въехать ровно один раз, поэтому

$$\sum_{i=1}^n x_{ij} = 1 \quad (j = 1, \dots, n). \quad (20)$$

Легко видеть, что этих условий недостаточно, чтобы описать допустимый маршрут коммивояжера (контур, или гамильтонов цикл). Например, две

Рис. 1.1. Ку-ку!

петли не соответствуют возможному маршруту коммивояжера, однако определяются условиями (18–20). Чтобы отбросить такие случаи, Таккер предложил ограничения (18–20) дополнить $n \cdot (n - 1)$ неравенствами вида:

$$u_i - u_j + nx_{ij} \leq n - 1 \quad (i = 1, \dots, n; j = 1, \dots, n; i \neq j), \quad (21)$$

где u_i — новые вещественные переменные.

Покажем, что если мы нашли решение, удовлетворяющее ограничениям (21), но не являющееся контуром, то нарушится по крайней мере одно из ограничений (21). В этом случае решение содержит не менее двух петель. Рассмотрим петлю, не проходящую через город с номером 1 и сложим все неравенства, отвечающие $x_{ij} = 1$, вдоль такой петли. Легко видеть, что в результате получим противоречие:

$$nk \leq (n - 1)k,$$

где k — число дуг в петле.

Теперь покажем, что для любого допустимого маршрута найдутся такие u_i , удовлетворяющие неравенствам (21). Положим

$$u_i = t, \quad \text{если город } i \text{ посещается на шаге } t, \quad (t = 1, \dots, n).$$

При $x_{ij} = 0$ неравенство (21) превращается в верное неравенство $u_i - u_j \leq n - 1$, а при $x_{ij} = 1$ имеем $u_i - u_j + nx_{ij} = t - (t + 1) + n = n - 1$.

Итак, задача коммивояжера состоит в минимизации функции

$$\sum_{j=1}^n x_{ij} a_{ij}$$

при ограничениях (18–21).

1.6. Задачи

- 1.1.** Выпукло ли множество точек $(x_1, x_2) \in \mathbf{R}^2$, для которых либо $x_1 \geq 0$, либо $x_2 \geq 0$?
- 1.2.** Доказать выпуклость множества $\text{Aff } M$, где M — произвольное множество точек.
- 1.3.** Доказать выпуклость множества $\text{Cone } M$, где M — произвольное множество точек.
- 1.4.** Доказать выпуклость множества $\text{Conv } M$, где M — произвольное множество точек.
- 1.5.** Дать геометрическую интерпретацию множествам $\text{Aff } M$, $\text{Cone } M$, $\text{Conv } M$, если $M = \{x_1, \dots, x_s\}$, $x_i \in \mathbf{R}^n$ ($i = 1, \dots, s$) и $1 \leq s \leq 4$, $1 \leq n \leq 3$.
- 1.6.** Доказать, что при аффинном преобразовании выпуклое множество переходит в выпуклое множество.
- 1.7.** Суммой множеств Y и Z называется множество $Y+Z = \{x : x = y + z, y \in Y, z \in Z\}$. Доказать, что сумма выпуклых множеств есть выпуклое множество.
- 1.8.** Пусть D — симметричная, положительно определенная матрица. Доказать выпуклость эллипсоида $x : x^\top D x \leq 1$. (Указание: использовать задачу 1.6 и пример 1.5.)
- 1.9.** Доказать, что сумма выпуклых функций — выпукла. Верно ли это для произведения выпуклых функций?
- 1.10.** Пусть $f(x)$ — выпуклая функция. Выпукла ли функция $|f(x)|$?
- 1.11.** Пусть $f(x)$ — выпуклая функция. Выпукло ли множество $\{x : f(x) \geq \alpha\}$?
- 1.12.** Пусть $f(x)$ и $g(x)$ — выпуклые функции. Можно ли утверждать, что функции $\min f(x), g(x)$ и $\max f(x), g(x)$ — выпуклы?
- 1.13.** Доказать, что множество минимумов в задаче выпуклого программирования выпукло.
- 1.14.** Показать оптимальность точки $x = 0$ для задачи:

$$\begin{cases} \max(x_1 - x_2) \\ x_1 - x_2 \leq 0, \\ 0 \leq x_2 \leq 1, \\ x_1 \geq 0. \end{cases}$$

Глава 2

Симплекс-метод

2.1. Числовой пример

Пример 2.1. Рассмотрим задачу линейного программирования:

$$\begin{cases} \max(2x_1 + x_2) \\ 3x_1 + 4x_2 + x_3 = 32, \\ 3x_1 + x_2 + x_4 = 17, \\ x_1 + x_5 = 5, \\ x_j \geq 0 \quad (j = 1, 2, \dots, 5). \end{cases} \quad (22)$$

Для ее решения сначала введем новую переменную

$$x_0 = 2x_1 + x_2.$$

Задача линейного программирования теперь заключается в максимизации линейной функции x_0 при ограничениях

$$\begin{cases} x_0 - 2x_1 - x_2 & = 0, \\ 3x_1 + 4x_2 + x_3 & = 32, \\ 3x_1 + x_2 & + x_4 = 17, \\ \boxed{x_1} & + x_5 = 5, \end{cases} \quad (23)$$

и требованиях неотрицательности:

$$x_j \geq 0 \quad (j = 1, 2, \dots, 5). \quad (24)$$

Роль переменной, помещенной в рамку, будет ясна в дальнейшем.

В системе линейных уравнений (62) уже выделены *базисные (связанные)* переменные x_0, x_3, x_4, x_5 и *небазисные (свободные)* переменные x_1, x_2 . Выразим базисные переменные через небазисные:

$$\begin{cases} x_0 = 2x_1 + x_2, \\ x_3 = 32 - 3x_1 - 4x_2, \\ x_4 = 17 - 3x_1 - x_2, \\ x_5 = 5 - x_1. \end{cases} \quad (25)$$

Обращая небазисные переменные в нуль, по формулам (25) получим частное решение $p = (0, 0, 32, 17, 5)$ и значение целевой функции $p_0 = 0$. Заметим, что это решение удовлетворяет требованиям неотрицательности (24) (это произошло из-за того, что правая часть в системе (62) неотрицательна). Таким образом, p — это план исходной задачи (57).

Очевидно, этот план не является оптимальным. Действительно, в выражении x_0 в (25) переменные x_1 и x_2 входят с положительными коэффициентами, поэтому значение целевой функции x_0 можно увеличить за счет увеличения x_1 и/или x_2 . Увеличивая x_1 и/или x_2 , мы, конечно, должны соблюсти условия (62, 24).

Попробуем увеличить x_0 , увеличивая x_1 , оставляя x_2 равным 0 и соблюдая ограничения (24), или эквивалентные им равенства (25). Выражение для x_3 в (25) позволяет увеличить x_1 до $32/2 = 16$ (напомним, что $x_3 \geq 0$). Из тех же соображений выражение для x_4 в (25) позволяет увеличить x_1 до $17/3$, а выражение для x_5 позволяет увеличить x_1 до 5. Так как

$$\min \left\{ 16, \frac{17}{3}, 5 \right\} = 5,$$

то на данном этапе x_1 может принять лишь значение 5.

Итак, $x_1 = 5, x_2 = 0$. Значения остальных переменных можно получить по формулам (25): $x_0 = 10, x_3 = 17, x_4 = 2, x_5 = 0$. Мы получили допустимый план и значение целевой функции на этом плане:

$$p = (5, 0, 17, 2, 0), \quad p_0 = 10. \quad (26)$$

Те же значения можно получить иным способом. Подставляя выражение для x_2 из (25) в уравнения (24) и приводя подобные, получим новую

систему, эквивалентную исходной:

$$\begin{cases} x_0 & - x_2 & & + 2x_5 = 10, \\ & 4x_2 + x_3 & & - 3x_5 = 17, \\ & \boxed{x_2} & + x_4 - 3x_5 & = 2, \\ x_1 & & & + x_5 = 5. \end{cases} \quad (27)$$

Конечно, более легкий (и грамотный) способ получения системы (27) из (24) состоит в применении одного шага гауссова исключения над элементом, заключенным в рамку. Теперь базисными переменными являются x_0, x_1, x_3, x_4 , а небазисными переменными являются x_2, x_5 . Как и раньше, выразим базисные переменные через небазисные. Получим:

$$\begin{cases} x_0 = 10 + x_2 - 2x_5, \\ x_3 = 17 - 4x_2 + 3x_5, \\ x_4 = 2 - x_2 + 3x_5, \\ x_1 = 5 - x_5. \end{cases} \quad (28)$$

Полагая небазисные переменные равными нулю, по формулам (27) получим тот же план и то же значение целевой функции, что и в (26).

В выражении x_0 через x_2 и x_5 из (28) переменная x_2 входит с положительным коэффициентом, поэтому значение x_0 может быть увеличено за счет увеличения x_2 до величины

$$\min \left\{ \frac{17}{4}, 2 \right\} = 2.$$

Чтобы получить новый план, сразу от системы (27) с помощью шага исключения Гаусса над элементом, заключенным в рамку, перейдем к системе

$$\begin{cases} x_0 & & + x_4 - x_5 = 12, \\ & x_3 - 4x_4 + \boxed{9x_5} = 9, \\ & x_2 & + x_4 - 3x_5 = 2, \\ x_1 & & + x_5 = 5, \end{cases} \quad (29)$$

откуда

$$\begin{cases} x_0 = 12 - x_4 + x_5, \\ x_3 = 9 + 4x_4 - 9x_5, \\ x_2 = 2 - x_4 + 3x_5, \\ x_1 = 5 - x_5. \end{cases} \quad (30)$$

Полагая небазисные переменные x_4, x_5 равными 0, получим новый план и значение целевой функции:

$$p = (5, 2, 9, 0, 0), \quad p_0 = 12.$$

Анализируя (31), приходим к выводу, что значение x_0 может быть еще увеличено за счет увеличения x_5 до величины, равной

$$\min \{1, 5\} = 1.$$

Делая в (29) шаг исключения Гаусса над элементом, заключенным в рамку, получим новую систему, эквивалентную исходной:

$$\begin{cases} x_0 & + \frac{1}{9}x_3 + \frac{5}{9}x_4 & = 13, \\ & \frac{1}{9}x_3 - \frac{4}{9}x_4 + x_5 & = 1, \\ & x_2 + \frac{1}{9}x_3 - \frac{1}{3}x_4 & = 5, \\ x_1 & - \frac{1}{9}x_3 + \frac{4}{9}x_4 & = 4, \end{cases}$$

откуда

$$\begin{cases} x_0 = 13 - \frac{1}{9}x_3 - \frac{5}{9}x_4, \\ x_5 = 1 - \frac{1}{9}x_3 + \frac{4}{9}x_4, \\ x_2 = 5 - \frac{1}{3}x_3 + \frac{1}{9}x_4, \\ x_1 = 4 + \frac{1}{9}x_3 - \frac{4}{9}x_4. \end{cases} \quad (31)$$

Полагая небазисные переменные x_3, x_4 равными 0, получим новый план и значение целевой функции:

$$p = (4, 5, 0, 0, 1), \quad p_0 = 13.$$

В выражении для x_0 из (31) переменные x_3 и x_4 входят с отрицательными коэффициентами, поэтому, так как $x_3 \geq 0, x_4 \geq 0$, то $x_0 \leq 13$. Итак, значение целевой функции не превосходит 13 и на плане $p = (4, 5, 0, 0, 1)$ значение 13 достигается, следовательно, этот план оптимальный.

Метод, с помощью которого мы решили данную задачу носит название *симплекс-метод*. Перейдем к его общему описанию.

2.2. Симплекс-метод в строчной форме

Рассмотрим каноническую ЗЛП (5-7). Воспользуемся матричной формой ее записи:

$$\min cx, \quad (32)$$

$$Ax = b, \quad (33)$$

$$x \geq 0. \quad (34)$$

Определение 2.2. Максимальная линейно независимая система столбцов матрицы называется ее (*столбцовой*) базой. Матрица, составленная из столбцов базы, называется *базисной подматрицей*.

Предположим, что ранг матрицы A равен числу ее строк n . Тогда множество столбцов образует базу тогда и только тогда, когда матрица B , составленная из этих столбцов квадратная и невырожденная, т.е. $\det B \neq 0$.

Определение 2.3. Пусть j_1, \dots, j_n — номера столбцов матрицы A , составляющих некоторую базу. Неизвестные x_{j_1}, \dots, x_{j_n} называются *базисными переменными* системы (33).

Обозначим

$$\mathcal{B} = \{j_1, \dots, j_n\}, \quad \mathcal{N} = \{1, 2, \dots, m\} \setminus \{j_1, \dots, j_n\}$$

— множества номеров, соответствующих базисным и небазисным переменным соответственно.

Пусть

$$A = (B, N), \quad x = \begin{pmatrix} x_{\mathcal{B}} \\ x_{\mathcal{N}} \end{pmatrix}, \quad c = (c_{\mathcal{B}}, c_{\mathcal{N}}),$$

тогда ЗЛП (32–34) примет вид:

$$\begin{cases} \max(c_{\mathcal{B}}x_{\mathcal{B}} + c_{\mathcal{N}}x_{\mathcal{N}}) \\ Bx_{\mathcal{B}} + Nx_{\mathcal{N}} = b, \\ x_{\mathcal{B}} \geq 0, \quad x_{\mathcal{N}} \geq 0. \end{cases}$$

Выразим базисные переменные $x_{\mathcal{B}}$ через небазисные:

$$x_{\mathcal{B}} = B^{-1}b - B^{-1}Nx_{\mathcal{N}}. \quad (35)$$

Определение 2.4. База B называется *допустимой*, если $B^{-1}b \geq 0$. Решение системы уравнений (33) вида $x_{\mathcal{B}} = B^{-1}b$, $x_{\mathcal{N}} = 0$ называется *опорным планом*, соответствующим базе B .

Пусть B — допустимая база, и p — соответствующий ей опорный план. Вычтем из целевой функции cx линейную комбинацию уравнений $Ax = b$ с коэффициентами $c_{\mathcal{B}}B^{-1}$. Получим равенство

$$cx = (c - c_{\mathcal{B}}B^{-1}A)x + c_{\mathcal{B}}B^{-1}b, \quad (36)$$

справедливое при любом удовлетворяющем системе (33) векторе x , положим $s_0 = c_{\mathcal{B}}B^{-1}b$, $s_j = c_j - c_{\mathcal{B}}B^{-1}a_j$, $j = 1, \dots, n$. Тогда равенство (36) переписется как

$$cx = s_0 + \sum_{j=1}^n s_j x_j, \quad (37)$$

Отметим, что коэффициенты s_j при $j \in \mathcal{B}$ в равенстве (37) равны 0, так как $c_{\mathcal{B}} - c_{\mathcal{B}}B^{-1}B = 0$. Отсюда следует, что $s_0 = cp$. Возможны три случая:

- 1) Все компоненты строки $s = (s_1, \dots, s_n)$ неотрицательны.
- 2) Найдется такое j , при котором $s_j < 0$ и $B^{-1}a_j \leq 0$.
- 3) Найдется такое j , при котором $s_j < 0$ и хотя бы одна компонента вектора $B^{-1}a_j$ положительна.

Теорема 2.5. *В первом случае опорный план p — оптимальное решение ЗЛП. Во втором случае значение целевой функции не ограничено снизу на множестве допустимых планов. В третьем случае существует допустимая база, отличная от B ровно одним столбцом, с соответствующим опорным планом q , для которого $cq \leq cp$.*

ДОКАЗАТЕЛЬСТВО. Если $s \geq 0$, то для любого плана x

$$cx = s_0 + \sum_{j=1}^n s_j x_j \geq s_0,$$

т.е. p — оптимальный план, и первое утверждение доказано. Если же найдется $l \in \mathcal{N}$ такое, что $s_l < 0$, то найдем вектор z , у которого $z_l = 1$, $z_j = 0$ при $j \in \mathcal{N}$, $j \neq l$, а остальные компоненты вычисляются по формуле $z_{\mathcal{B}} = -B^{-1}a_l$. Тогда

$$Az = \sum_{j=1}^n a_j z_j = a_l - A_j(A_j^{-1}a_l) = 0 \quad \text{и} \quad cz = \sum_{j \in \mathcal{B}} c_j z_j + \sum_{j \in \mathcal{N}} c_j z_j = c_l - c_{\mathcal{N}}A_{\mathcal{N}}^{-1}a_l = s_l.$$

При произвольном α рассмотрим вектор $p' = p + \alpha z$. Для него $Ap' = Ap + \alpha Az = Ap = b$ и $cp' = cp + \alpha pz = s_0 + \alpha s_l$. Отсюда следует второе из доказываемых утверждений, так как в этом случае $p' \geq 0$ при любом $\alpha \geq 0$ и, следовательно, p' является планом рассматриваемой ЗЛП, а $cp' \rightarrow \infty$ при $\alpha \rightarrow \infty$, так как $s_l < 0$.

Третий случай отличается от второго тем, что некоторые компоненты вектора z являются отрицательными. Обозначим наибольшее значение величины α , при которой вектор $p + \alpha z$ имеет неотрицательные компоненты, через β . Имеем $\beta = \min_{z_i < 0} -p_i/z_i$. Ясно, что $q = p + \beta z$ является допустимым планом и

$$cq \leq cp. \quad (38)$$

Покажем, что вектор q является опорным планом. Пусть ν — номер, при котором $\beta = -p_\nu/z_\nu$. Обозначим через I множество, получающееся из \mathcal{B} заменой ν на l , и покажем, что множество A_I является базой, которой соответствует опорный план q . Действительно, матрица $B^{-1}A_I$ отличается от единичной матрицы только одним столбцом, равным $B^{-1}a_l = -z_{\mathcal{B}}$, а ее определитель равен $-z_\nu \neq 0$. Таким образом, матрица $B^{-1}A_I$, а, следовательно, и матрица A_I являются невырожденными. Так как номера ненулевых компонент вектора q принадлежат множеству I , следовательно, базе A_I соответствует опорный план q . ■

Поставим в соответствие базе A матрицу размеров $(m+1) \times (n+1)$, нумерация строчек и столбцов которой начинается с нуля:

$$\left(\begin{array}{c|c} -s_0 & s \\ \hline A_1^{-1}b & A_1^{-1}A \end{array} \right) \quad (39)$$

Симплекс-таблицу будем рассматривать как матрицу размерами , нумерация строчек и столбцов которой начинается с нуля. Непосредственной проверкой легко убедиться, что матрица S^I равна произведению матриц

$$F^I = \left(\begin{array}{c|c} 1 & -c_I A_I^{-1} \\ \hline 0 & A_I^{-1} \end{array} \right) \quad \text{и} \quad S = \left(\begin{array}{c|c} 0 & c \\ \hline b & A \end{array} \right),$$

имеющих размеры $(m+1) \times (m+1)$ и $(m+1) \times (n+1)$, соответственно. Пусть B — некоторая база. Из соотношений $S^I = F^I S$, $S^{\mathcal{B}} = F^{\mathcal{B}} S$ выводим равенство $S^{\mathcal{B}} = (F^{\mathcal{B}}(F^I)^{-1})S^I$. По правилу умножения матриц из последнего равенства следует, что строки матрицы $S^{\mathcal{B}}$ являются линейными комбинациями строк матрицы S^I . Пусть база B получается из

базы A_I в результате замены индекса j на индекс i , т.е. $\mathcal{B} = I \setminus \{j\} \cup \{i\}$. Обозначим через $\mu(j)$ порядковый номер индекса j во множестве I . Из сопоставления матриц $S_{\mathcal{B}}^I$ и S_j^I отличающихся друг от друга только $\mu(j)$ -ым столбцом, равным $e_{\mu(j)}$ и S_i^I , соответственно, находим коэффициенты в этих комбинациях

$$S_{k*}^{\mathcal{B}} = \begin{cases} \frac{1}{s_{\mu(j)i}^I} \cdot s_{\mu(j)*}^I & \text{при } k = \mu(j) \\ s_{k*}^I - \frac{s_{ki}^I}{s_{\mu(j)i}^I} \cdot s_{\mu(j)*}^I & \text{при } k \neq \mu(j) \end{cases} \quad (40)$$

Для формального доказательства формул (40) приведем ряд выкладок. Легко проверить, что

$$(F^{\mathcal{B}})^{-1} = \begin{pmatrix} 1 & c_j \\ 0 & B \end{pmatrix} \quad \text{и} \quad F^I (F^{\mathcal{B}})^{-1} = \begin{pmatrix} 1 & c_{\mathcal{B}} - c_I A_I^{-1} B \\ 0 & A_I^{-1} B \end{pmatrix} = \begin{pmatrix} 1 & s_{\mathcal{B}}^I \\ 0 & s_{\mathcal{B}}^I \end{pmatrix}.$$

Из последнего равенства видно, что матрица $F^I (F^{\mathcal{B}})^{-1}$ отличается от единичной только $\mu(j)$ -ым столбцом, который равен S_{*i}^I . Но тогда обратная к ней $F^I (F^{\mathcal{B}})^{-1}$ отличается от единичной матрицы только $\mu(j)$ -ым столбцом, равным $(-s_{0i}^I/s_{\mu(j)i}^I \dots -s_{\mu(j)-1i}^I/s_{\mu(j)i}^I 1/s_{\mu(j)i}^I -s_{\mu(j)+1i}^I/s_{\mu(j)i}^I \dots -s_{mi}^I/s_{\mu(j)i}^I)$. Отсюда и из равенства $S^{\mathcal{B}} = (F^I (F^{\mathcal{B}})^{-1})^{-1} S^I$ вытекают формулы (40).

Симплекс-таблица (39) содержит все необходимые данные для перехода к следующей базе. Дадим описание симплекс-метода.

Алгоритм 1. [Симплекс-метод. Строчечная форма.]

Шаг 1. Построить симплекс-таблицу S^I , соответствующую начальной допустимой базе I .

Шаг 2. Найти $i \in \{1, 2, \dots, n\}$, для которого s_{0i}^I . Если таких нет i , то стоп: текущая база оптимальна.

Шаг 3. Найти индекс j , при котором s_{ij}^I и достигается минимум

$$\frac{s_{j0}^I}{s_{ij}^I} = \min_{k/s_{ki}^I > 0} \frac{s_{kl0}^I}{s_{kj}^I}.$$

Если при всех k имеет место неравенство $s_{kj}^I \leq 0$, то стоп; оптимального плана не существует, так как линейная форма не ограничена снизу на множестве допустимых планов.

Шаг 4. Заменить элемент $\mu^{-1}(j)$ в базе I на i . Пересчитать симплекс-таблицу по формулам (40), и вернуться на шаг 2.

Пример 2.6. Для примера решим строчечным симплекс-методом ЗЛП

$$\begin{aligned} & \max(x_1 + x_2) \\ & \begin{cases} 2x_1 + x_2 + x_3 = 6, \\ x_1 + 2x_2 + x_4 = 6, \\ x_j \geq 0 \quad (j = 1, \dots, 4). \end{cases} \end{aligned}$$

В качестве начальной допустимой базы можно положить $\{3, 4\}$, ей будет соответствовать следующая симплекс-таблица:

$$\left(\begin{array}{c|cccc} 0 & -1 & -1 & 0 & 0 \\ \hline 6 & \boxed{2} & 1 & 1 & 0 \\ 6 & 1 & 2 & 0 & 1 \end{array} \right)$$

На втором шаге i можно выбрать, как равным 1, так и равным 2. Для определенности положим $i = 1$. На третьем шаге находим $j = 1$, то есть меняем первый элемент базы. Следующая база равна $\{1, 4\}$. Вычислим симплекс-таблицу новой базы по формулам (40). Для этого к нулевой строке прибавим первую строку, умноженную на $1/2$, из второй строки вычтем первую умноженную на $1/2$ и первую строку умножим на $1/2$. В результате получим симплекс-таблицу

$$\left(\begin{array}{c|cccc} 3 & 0 & -1/2 & 1/2 & 0 \\ \hline 3 & 1 & 1/2 & 1/2 & 0 \\ 3 & 0 & \boxed{3/2} & -1/2 & 1 \end{array} \right)$$

Полученная база не является оптимальной, о чем свидетельствует наличие отрицательного элемента в нулевой строке. На втором и третьем шагах симплекс-метода находим $i = 2$ и $j = 2$. Следовательно, следующая база равна $\{1, 2\}$, ей соответствует симплекс-таблица

$$\left(\begin{array}{c|cccc} 4 & 0 & 0 & 1/3 & 1/3 \\ \hline 2 & 1 & 0 & 2/3 & -1/3 \\ 2 & 0 & 1 & -1/3 & 2/3 \end{array} \right)$$

Эта таблица — оптимальна. Опорный план, соответствующий этой базе, равен $(2, 2, 0, 0)^T$.

Переход от базы к следующей базе называется *итерацией* симплекс-метода. Столбец i и строка j , получаемые на шаге 2 и 3 симплекс-метода, называются *ведущими*. Элемент симплекс-таблицы, стоящий на пересечении ведущей строки и ведущего столбца называется *ведущим*.

Определение 2.7. База A_I называется *невырожденной*, если все компоненты опорного плана с номерами из I , отличны от нуля. Задача линейного программирования называется *невырожденной*, если все ее допустимые базы являются невырожденными.

В случае невырожденной базы величина β строго больше нуля, поэтому неравенство (??) — строгое. ает схему последовательного улучшения опорного плана. Так как число баз конечно, то в случае невырожденной ЗЛП получаем конечный алгоритм ее решения. Для перехода от текущей базы A_I к следующей достаточно вычислить вектор $c_{\bar{I}} - c_I A_I^{-1} A_{\bar{I}}$ и матрицу $A_{\bar{I}}^{-1} A_{\bar{I}}$.

Теорема 2.8. Симплекс-метод, примененный к невырожденной задаче линейного программирования, конечен.

ДОКАЗАТЕЛЬСТВО. Число различных баз задачи линейного программирования не превосходит числа

$$\frac{n!}{(n-m)! m!}$$

всех m -элементных подмножеств n -элементного множества. Значение целевой функции на каждой итерации симплекс-метода уменьшается, поэтому повторение баз невозможно. Таким образом, через конечное число итераций симплекс-метод должен остановиться. ■

2.3. Зацикливание и способы защиты от него

2.3.1. Зацикливание

Выбор элемента, вводимого в базу, и элемента, выводимого из базы, в симплекс-методе определен, вообще говоря, неоднозначно. Желательно

конкретизировать этот выбор так, чтобы любая ЗЛП решалась бы как можно быстрее. Однако надеяться на точное решение такой «сверхзадачи» в настоящее время трудно, и поэтому приходится использовать такие приемы, от которых требуется, чтобы, во-первых, любая ЗЛП решалась за конечное число итераций, и, во-вторых, это число итераций по каким-то эвристическим соображениям было бы «не очень большим» хотя бы в большинстве случаев.

Пусть $Q = (q_{ij})$ — симплекс-таблица, соответствующая некоторой допустимой базе. Обычно применяются следующие критерии для выбора номера вводимого в базу (шаг 2 симплекс-метода).

α_1 Выбираем первый встретившийся индекс l , удовлетворяющий условию $q_{0l} < 0$.

α_2 Выбираем индекс l , с наименьшим значением q_{0l} .

α_3 Выбираем l с наименьшим значением величины θq_{0l} .

Первый из этих критериев — самый простой. Третий — дает наибольшее увеличение целевой функции за одну итерацию симплекс-метода. Второй занимает промежуточное положение между первым и третьим, он обещает наибольшее увеличение целевой функции при увеличении на 1.

Простое правило выбора элемента выводимого из базы при выбранном i (шаг 3 симплекс-метода) можно сформулировать так:

β_1 Выбираем первый встретившийся индекс j , на котором достигается минимум отношения

$$\frac{q_{j0}}{q_{jl}}.$$

Пример 2.9. Начиная с допустимой базы $\{2, 3, 4, 5\}$ применить симплекс-метод с критериями α_1 и β_1 к ЗЛП

$$\begin{cases} \max(-600x_1 + 75x_6 - 500x_7 + 2x_8) \\ 9x_1 + x_2 + 1/4x_6 - 2x_7 - 1/25x_8 = 0, \\ 3x_1 + x_3 + 1/2x_6 - 3x_7 - 1/50x_8 = 0, \\ x_4 + x_8 = 1, \\ -225/2x_1 + x_5 - 25x_6 + 200x_7 + x_8 = 0, \\ x_j \geq 0 \quad (j = 1, 2, \dots, 8). \end{cases}$$

$$\begin{aligned}
& \mathcal{B}(4) = \{8, 7, 4, 5\}, \\
Q(4) = & \left(\begin{array}{c|ccccccccc}
0 & -300 & -200 & 300 & 0 & 0 & 25 & 0 & 0 \\
\hline
0 & -525/2 & -75/2 & 25 & 0 & 0 & 25/8 & 0 & 1 \\
0 & \boxed{3/4} & 1/4 & -1/2 & 0 & 0 & -3/16 & 1 & 0 \\
1 & 525/2 & 75/2 & -25 & 1 & 0 & -\frac{25}{8} & 0 & 0 \\
0 & 0 & -25/2 & 75 & 0 & 1 & 75/8 & 0 & 0
\end{array} \right), \\
& \mathcal{B}(5) = \{8, 1, 4, 5\}, \\
Q(5) = & \left(\begin{array}{c|ccccccccc}
0 & 0 & -100 & 100 & 0 & 0 & -50 & 400 & 0 \\
\hline
0 & 0 & \boxed{50} & -150 & 0 & 0 & -125/2 & 350 & 1 \\
0 & 1 & 1/3 & -2/3 & 0 & 0 & -1/4 & 4/3 & 0 \\
1 & 0 & -50 & 150 & 1 & 0 & 125/2 & -350 & 0 \\
0 & 0 & -25/2 & 75 & 0 & 1 & 78/8 & 0 & 0
\end{array} \right), \\
& \mathcal{B}(6) = \{2, 1, 4, 5\}, \\
Q(6) = & \left(\begin{array}{c|ccccccccc}
0 & 0 & 0 & -200 & 0 & 0 & -175 & 1100 & 2 \\
\hline
0 & 0 & 1 & -3 & 0 & 0 & -5/4 & 7 & 1/50 \\
0 & 1 & 0 & \boxed{1/3} & 0 & 0 & 1/6 & -1 & -1/150 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 75/2 & 0 & 1 & -25/4 & 175/2 & 1/4
\end{array} \right).
\end{aligned}$$

Далее $\mathcal{B}(7) = \mathcal{B}(1)$ и, следовательно, произошло так называемое за-
цикливание и процесс бесконечен.

Не спасает и использование других критериев α_2 и α_3 выбора i , так
как в рассматриваемом примере они приводят к одному и тому же ре-
зультату.

Вычислительные процедуры, гарантирующие от зацикливания, разо-
браны ниже.

2.3.2. Лексикографический метод

Определение 2.10. Говорят, что вектор p лексикографически положителен (обозначение $p \succ 0$), если его первая отличная от нуля компонента положительна. Вектор p лексикографически больше вектора q (обозначение $p \succ q$), или вектор q лексикографически меньше вектора p (обозначение $q \prec p$), если $p - q \succ 0$.

Сформулируем теперь лексикографический критерий выбора строки в строчечной форме симплекс-метода.

β_2 Выбирай j , при котором $s_{ji}^I > 0$ и

$$\frac{1}{s_{ji}^I} \cdot s_j^I \succ \frac{1}{s_{ki}^I} \cdot s_k^I$$

для всякого k , такого, что $s_{ki}^I > 0$.

Такой выбор j , будем записывать формулой

$$\frac{1}{s_{ji}^I} \cdot s_j^I = \text{lexmin}_{s_{ki}^I > 0} \frac{1}{s_{ki}^I} \cdot s_k^I$$

Замечание 2.11. Так как матрица S не содержит одинаковых строк, то j по критерию β_2 определяется однозначно.

Определение 2.12. Симплексную таблицу будем называть лексикографически допустимой, если все ее строки, кроме нулевой, лексикографически больше 0.

Лемма 2.13. Свойство лексикографической допустимости симплекс-таблицы сохраняется, если на шаге 3 симплекс-метода для выбора ведущей строки используется критерий β_2 .

Доказательство. Пусть за одну итерацию симплекс-метода осуществляется переход от базы I к базе L . Пересчет элементов симплекс-таблицы ведется по формулам (40). Ведущая строка с номером j пересчитывается по формуле

$$s'_{j*} = (1/s_{ji})s_{j*},$$

и, следовательно, остается лексикографически положительной. Любая другая строка k пересчитывается по формуле

$$s'_{k*} = s_{k*} - \frac{s_{ki}}{s_{ji}} \cdot s_{j*}.$$

Если $s_{ki} \leq 0$, то k -ая строка лексикографически больше 0 как сумма лексикографически положительных строк.

Если $s_{ki} > 0$, то преобразуем формулу вычисления k -ой строки к виду

$$s'_{ki} = s_{ki} \cdot \left(\frac{1}{s_{ki}} \cdot s_{k*} - \frac{1}{s_{ji}} \cdot s_{j*} \right).$$

В скобках стоит лексикографически положительный вектор (по способу выбора j), который умножается на положительное число. Таким образом, в любом случае строка k ($k > 0$) новой симплекс-таблицы лексикографически положительная. ■

Определение 2.14. Симплекс-метод, в котором номер направляющей строки j выбирается по критерию β_3 , будем называть *лексикографическим симплекс-методом*.

Теорема 2.15. При любом из способов $\alpha_1, \alpha_2, \alpha_3$ выбора направляющего столбца, лексикографический симплекс-метод, примененный к лексикографически допустимой таблице, конечен.

Доказательство. По лемме изложенной выше все таблицы, появляющиеся в лексикографическом симплекс-методе, являются лексикографически положительными. На каждой итерации симплекс-метода к нулевой строке прибавляется лексикографически положительная строка. Таким образом, при переходе от базы к базе происходит лексикографическое увеличение нулевой строки симплекс-таблицы, и, следовательно, заикливание невозможно. ■

Пример 2.16. Проиллюстрируем лексикографический симплекс-метод на ЗЛП из примера 2.2.1. Будем выбирать номер ведущего элемента по критериям α_1, β_2 . Начальная база равна $\mathcal{B}_1 = \{2, 3, 4, 5\}$. Далее, как и выше, вместе с базой будем приводить соответствующую ей симплекс-таблицу.

2.4. ПОЛУЧЕНИЕ НАЧАЛЬНОГО ДОПУСТИМОГО ОПОРНОГО ПЛАНА 39

Пусть r — максимальный номер, который был исключен из какой-то из перечисленных баз, для определенности, скажем из базы \mathcal{B} . Обозначим через t номер, вводимый в базу \mathcal{B} . Очевидно, на какой-то итерации он добавится. Пусть r добавляется в базу L . Определим вектор z , компоненты которого вычисляются по формулам: $z_{\mathcal{B}} = -B^{-1}a_t$, $z_t = 1$, а остальные равны нулю. По правилу выбора ведущего столбца $cz = s_{0t}^{\mathcal{B}} < 0$. С другой стороны,

$$cz = s_{0*}^L z = \sum_{i \in L} s_{0i}^L z_i.$$

Если $i \notin \mathcal{B}$, $i \neq t$, то $z_i = 0$ и $s_{0i}^L z_i = 0$.

Если $i < r$, то по правилу выбора ведущего элемента $s_{0i}^L \geq 0$ и $z_i \geq 0$ (здесь либо $i = t$, либо $i \in \mathcal{B}$), т.е. $z_{0i}^L z_i \geq 0$.

Если $i = r$, то $z_r < 0$ и $s_{0r}^L < 0$, т.е. $z_{0i}^L z_i \geq 0$.

И, наконец, если $i > r$ и $i \in \mathcal{B}$, то по определению r , номер i принадлежит всем базам, и, следовательно, базе L , а тогда $s_{0i}^L = 0$.

Тем самым установлена неотрицательность всех слагаемых $s_{0i}^L z_i$, а значит и их суммы cz , что противоречит полученному ранее. К данному противоречию привело предположение о закливание симплекс-метода с правилом Бленда выбора ведущего элемента. ■

Пример 2.18. Проиллюстрируем симплекс-метод с правилом Бленда выбора ведущего элемента на ЗЛП из примера 2.2.1. Начальная база равна $\mathcal{B}_1 = \{2, 3, 4, 5\}$. Далее, как и выше, будут построены базы $\mathcal{B}_2 = \{6, 3, 4, 5\}$, $\mathcal{B}_3 = \{6, 7, 4, 5\}$, $\mathcal{B}_4 = \{8, 7, 4, 5\}$. Начиная с этого момента, вместе с базой будем приводить соответствующую ей симплекс-таблицу.

Последняя база является оптимальной, и соответствующий ей план равен $(0, \frac{3}{100}, 0, 0, 0, \frac{1}{25}, 0, 1)^T$.

2.4. Получение начального допустимого опорного плана

Для начала работы симплекс-метода требуется начальная допустимая база. Однако, в общем случае, непросто найти допустимую базу ЗЛП. Может оказаться, например, что допустимых баз вообще нет. В литературе описано большое количество способов получения начальной допустимой базы. Наиболее часто встречаемый подход заключается в сведении задачи получения начальной допустимой базы к некоторой задаче линейного программирования, с известной начальной допустимой базой,

по решению которой находится допустимая база исходной задачи. Примером такого подхода является метод искусственного базиса, описанный ниже.

В канонической задаче линейного программирования (32–34) все компоненты вектора b можно считать неотрицательными. Действительно, иначе умножим уравнения (33), с отрицательными правыми частями на -1 . Введем искусственные переменные y_i ($i = 1, 2, \dots, m$), которые можно интерпретировать как «невязки» между правыми и левыми частями в (33), и рассмотрим ЗЛП

$$\min \sum_{i=1}^m y_i, \quad (41)$$

$$y + Ax = b, \quad (42)$$

$$y \geq 0, \quad x \geq 0. \quad (43)$$

В качестве начальной базы задачи (41–43) можно положить множество $I = \{1, 2, \dots, m\}$. Симплекс-таблица, соответствующая базе I равна

$$\left(\begin{array}{c|cc} \sum_{i=1}^m b_i & 0 & -\sum_{i=1}^m a_{i*} \\ b & E & A \end{array} \right)$$

Эта таблица является лексикографически допустимой. Значение целевой функции в ЗЛП (41–43) ограничено снизу нулем. Следовательно, применив к ней лексикографический симплекс-метод, через конечное число итераций, получим базу, соответствующую оптимальному плану. Возможно два случая.

- 1) Значение целевой функции на оптимальном плане ЗЛП (41–43) строго больше 0.
- 2) Значение целевой функции на оптимальном плане ЗЛП (41–43) равно 0.

В первом случае не существует допустимой базы ЗЛП (32–34). Действительно, если система уравнений и неравенств (33–34) имеет допустимое решение x^* , то план $(0, x^*)$ является допустимым для задачи линейного программирования (41–43), и значение целевой функции на нем равно 0.

2.4. ПОЛУЧЕНИЕ НАЧАЛЬНОГО ДОПУСТИМОГО ОПОРНОГО ПЛАНА 41

Во втором случае существует допустимая база ЗЛП (32–34). Причем эту базу легко построить по оптимальной базе \mathcal{B} ЗЛП (41–43). Обозначим через S симплекс-таблицу, соответствующую базе \mathcal{B} . Значение переменных на оптимальном плане равно 0. Если база \mathcal{B} невырожденная, то столбцы с номерами $1, 2, \dots, m$ не являются базисными, и вычеркнув их, получим допустимую базисную матрицу для ЗЛП (32–34). Если же \mathcal{B} — вырожденная база, то может случиться так, что некоторые из номеров $1, 2, \dots, m$ содержатся в ней. Допустим, это номер i . Рассмотрим строку $\mu(i)$ матрицы S . Известно, что $s_{\mu(i),0} = 0$ и $s_{\mu(i),i} = 1$.

Если найдется такое $j > m$, что $s_{\mu(i),j} \neq 0$, то сделаем итерацию симплекс-метода с ведущим элементом $s_{\mu(i),j}$. В результате получится допустимая симплекс-таблица, и номер i будет исключен из базы.

Если же $s_{\mu(i),j} = 0$ для всех $j > m$, то ранг матрицы A меньше m , то есть система (33) избыточна, так как она содержит уравнение, являющееся линейной комбинацией ее остальных уравнений. Рассматриваемый метод позволяет обнаружить эту избыточность и уменьшить число лишних уравнений на единицу вычеркиванием $\mu(i)$ -ой строки и i -го столбца.

Итак, в обоих случаях число базисных столбцов, соответствующих искусственным переменным, уменьшится на единицу. Повторяя описанный процесс, получим таблицу, сохраняющую свойство допустимости и не содержащую столбцов, соответствующих искусственным переменным. Результирующая база будет являться допустимой для ЗЛП (32–34). Симплекс-таблица для ЗЛП (32–34), соответствующая этой базе, будет отличаться от уже полученной, только нулевой строкой. Для получения допустимой симплекс-таблицы для ЗЛП (32–34) достаточно на место нулевой строки поместить целевую функцию (32) и провести итерации симплекс-метода, беря в качестве ведущих базисные столбцы. В результате получим допустимую, но не обязательно лексикографически допустимую симплекс-таблицу для ЗЛП (32–34). Привести ее к лексикографически допустимой симплекс-таблице можно, изменив нумерацию переменных, например, поставив столбцы полученного допустимого базиса на первые места.

симплекс-таблица не оптимальна¹.

$$Q(3) = \left(\begin{array}{c|ccccccc} & & & \mathcal{B}(3) = \{2, 4, 5\}, & & & & \\ \hline 0 & 1/3 & 0 & 4/3 & 0 & 0 & 0 & 0 \\ 0 & 5/9 & 0 & -1/9 & 0 & 1 & -4/3 & 4/3 \\ 0 & 2/3 & 1 & -1/3 & 0 & 0 & 0 & 0 \\ 1 & 4/9 & 0 & 1/9 & 1 & 0 & 1/3 & 5/3 \end{array} \right).$$

Таблица $Q(3)$ оптимальна. Так как $q_{00} = 0$, то исходная система совместна. Так как $q_{2j} = 0$ ($j = 4, 5, 6, 7$), то вторая строка избыточна. Вычеркивая вторую строку и столбцы, соответствующие искусственным переменным, и добавив целевую функцию исходной задачи, получим таблицу:

$$\left(\begin{array}{c|cccc} & & & \mathcal{B} = \{1, 2\}, & \\ \hline -1 & 0 & 0 & -1 & 4 \\ 0 & 0 & 1 & -4/3 & 4/3 \\ 1 & 1 & 0 & 1/3 & 5/3 \end{array} \right).$$

Проведя итерации симплекс-метода с ведущими базисными столбцами, получим допустимую симплекс-таблицу:

$$\left(\begin{array}{c|cccc} & & & & \\ \hline -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & -4/3 & 4/3 \\ 1 & 1 & 0 & 1/3 & 5/3 \end{array} \right).$$

Таблица уже оптимальна, поэтому сразу получаем оптимальный целевой вектор $p = (1, 0, 0, 0)^T$ и значение целевой функции $p_0 = 1$.

2.5. Задачи

2.1. Показать, что если система векторов, соответствующих положительным компонентам допустимого плана, линейно зависима, то существует допустимый план с меньшим числом положительных компонент.

2.2. Решить ЗЛП:

¹ Тем не менее дальнейшую оптимизацию можно прекратить и приступить к исключению из базиса столбцов с номерами меньшими чем m .

$$\max(-x_1 + 2x_2 + 4x_3)$$

$$\begin{cases} x_1 - x_2 - x_3 = 1, \\ x_1 + x_2 + x_3 = 3, \\ x_1, x_2, x_3 \geq 0; \end{cases}$$

$$\max(x_1 + x_2 + x_3)$$

$$\begin{cases} -2x_1 + x_2 + x_3 = 4, \\ x_1 - 2x_2 + x_3 = 1, \\ x_1, x_2, x_3 \geq 0; \end{cases}$$

$$\max(x_1 + x_2 + x_3 - 2x_4 + x_5 - 2x_6)$$

$$\begin{cases} x_1 - x_2 + x_3 - x_4 + x_5 - x_6 = 7, \\ 2x_1 + x_2 - 2x_3 - 3x_4 + 2x_5 + 3x_6 = 0, \\ 3x_1 + 2x_2 - x_3 - 4x_4 + 3x_5 + 2x_6 = 10, \\ x_j \geq 0, \quad (j = 1, \dots, 6); \end{cases}$$

$$\max(-5x_1 - x_2 + 3x_3 + 4x_4)$$

$$\begin{cases} -3x_1 + x_2 + x_3 + x_4 - x_5 - x_6 = 6, \\ x_1 - 3x_2 + x_3 + x_4 - x_5 - x_6 = 2, \\ x_1 + x_2 - 3x_3 + x_4 - x_5 - x_6 = 2, \\ -x_1 - x_2 - x_3 + 3x_4 - 3x_5 - x_6 = 10, \\ x_j \geq 0, \quad (j = 1, \dots, 6); \end{cases}$$

$$\max(-x_1 - 2x_2 + 3x_3 - 2x_4)$$

$$\begin{cases} -2x_1 - x_2 + 3x_3 + x_4 = 4, \\ x_1 + 2x_2 + 3x_3 + 2x_4 = 6, \\ 3x_1 - x_2 - 2x_3 + x_4 \geq 2, \\ 5x_1 + 3x_2 + x_3 \leq 6, \\ -2x_1 + x_2 - 3x_3 - 2x_4 \leq 4, \\ x_1 \geq 0, \\ x_2 \geq 0. \end{cases}$$

2.6. Столбцовая форма

При использовании какого-либо алгоритма возникает проблема минимизации объема вычислений. Как отмечалось ранее, проблема минимизации количества итераций симплекс-метода является сложной. Основ-

ное внимание при оптимизации симплекс-метода уделяется уменьшению объема вычислений необходимых для итерации симплекс-метода. Число элементарных арифметических операций, необходимых для одной итерации симплекс-метода в строчечной форме не превосходит $2(m+1)(n+1)$.

Рассмотрим применение общей схемы симплекс-метода, описанного в 2.1, к стандартной задаче линейного программирования (5–7). Конечно, можно стандартную ЗЛП привести к канонической, как это описано в 1.3, при этом трудоемкость одной итерации будет равна $(m-n+1)(m+n+1)$, где m число неравенств в (5), а n — число переменных. При значениях m превышающих n более чем в 2 раза основное место в симплекс-таблице занимает единичная матрица размером $m \times m$, расположенная в базисных столбцах. Тогда как для итерации симплекс-метода необходимо знание коэффициентов симплекс-таблицы как раз в небазисных столбцах. Итерация симплекс-метода, по сути, состоит в том, что переменные с номерами из столбцовой базы выражаются через остальные переменные по формулам (35). Для удобства проведения рассуждений представим стандартную задачу линейного программирования в форме общей ЗЛП:

$$\min cx,$$

$$\hat{A}x \leq \hat{b}.$$

Здесь матрица

$$\hat{A} = \begin{pmatrix} A \\ -E \end{pmatrix}$$

размером $(m+n) \times n$ является матрицей коэффициентов при неизвестных в неравенствах (6), (7), причем неравенства (7) записаны в виде $-x \leq 0$, а вектор $\hat{b} = (b, 0)$ размерности $m+n$ — столбец свободных членов в неравенствах (6, 7).

Обозначим через \tilde{A} матрицу E, A . Эта матрица возникает при сведении стандартной ЗЛП к канонической (12). Пусть I столбцовая база матрицы \tilde{A} , \bar{I} — множество номеров остальных столбцов. Обозначим через y переменные в канонической ЗЛП (12) с номерами из \bar{I} . Имеет место равенство

$$\hat{A}_{\bar{I}^*} x + y = b. \quad (44)$$

Лемма 2.20. Если I — столбцовая база матрицы \tilde{A} , то \bar{I} — строчечная база \hat{A} .

ДОКАЗАТЕЛЬСТВО. Разобьем множество I на два непересекающихся подмножества

$$I_1 = \{i : i \in I, i \leq m\} \quad \text{и} \quad I_2 = \{i : i \in I, i > m\}.$$

Обозначим через N_k множество чисел $\{1, \dots, k\}$. Положим

$$I_3 = \{(i - m) : i \in I_2\}.$$

То, что множество I является базой, равносильно невырожденности матрицы $(E, A)_{*I}$. Так как $(E, A)_{*I} = (E_{*I_1}, A_{I_3})$, то разложив ее по первым столбцам, принадлежащим единичной матрице, получим, с точностью до знака, определитель матрицы $A_{N_m \setminus I_1, I_3}$. Аналогично, разложив определитель матрицы

$$\begin{pmatrix} A_{N_m \setminus I_1} \\ -E_{N_n \setminus I_3} \end{pmatrix}$$

по последним строкам, получим, с точностью до знака, определитель матрицы $A_{N_m \setminus I_1, I_3}$. Невырожденность матрицы $\hat{A}_{\bar{I}}$ означает, что \bar{I} — строчечная база \hat{A} . ■

По только что доказанной лемме, в равенстве (44) матрица $\hat{A}_{\bar{I}}$ — невырожденная. Выразим x через y . Опорный план, соответствующий базе I , получается при $y = 0$, и, значит, равен $\hat{A}_{\bar{I}}^{-1}b$. Обозначим его через p^I . Поставим в соответствие строковой базе I симплекс-таблицу

$$\left(\begin{array}{c|c} -cp^I & -x\hat{A}_{\bar{I}}^{-1} \\ \hat{b} - \hat{A}p^I & -\hat{A}\hat{A}_{\bar{I}}^{-1} \end{array} \right).$$

В некоторых случаях полезно рассматривать симплекс-таблицу T^I в виде матрицы размеров $(m+n+1) \times (n+1)$, нумерация строк и столбцов которой начинается с нуля. Непосредственной проверкой нетрудно убедиться, что матрица T^I равна произведению матриц

$$T^M = \begin{pmatrix} 0 & c \\ \hat{b}\hat{A} & \end{pmatrix} \quad \text{и} \quad Q^I = \begin{pmatrix} 1 & 0 \\ -p^I & -\hat{A}\hat{A}_{\bar{I}}^{-1} \end{pmatrix},$$

имеющих размеры $(m+n+1) \times (n+1)$ и $(n+1) \times (n+1)$ соответственно.

Обозначим через \mathcal{B} базу, получаемую из базы I в результате замены ее j -го элемента, равного i , на k . Из формул $T^I = T^M Q^I$ и $T^{\mathcal{B}} = T^M Q^{\mathcal{B}}$

вытекает равенство $T^j = T^I(Q^I)^{-1}Q^{\mathcal{B}}$. Поскольку матрица $T^{\mathcal{B}}$ получается из матрицы T^I путем умножения справа на некоторую матрицу, то по правилу перемножения матриц, столбцы матрицы $T^{\mathcal{B}}$ являются линейными комбинациями столбцов матрицы T^I . Из сопоставления матриц T^I и $T^{\mathcal{B}}$, отличающихся друг от друга только j -ой строчкой, равной t_k^I и $-e_j$ соответственно, находим коэффициенты в этих комбинациях:

$$t_l^{\mathcal{B}} = \begin{cases} -\frac{1}{t_{kj}^I} \cdot t_j^I, & \text{при } l = j, \\ t_l^I - \frac{t_{kl}^I}{t_{kj}^I} \cdot t_j^I, & \text{при } l \neq j \end{cases} \quad (45)$$

Для формального обоснования формул (45) проведем ряд выкладок. Легко проверить, что

$$(Q^{\mathcal{B}})^{-1} = \begin{pmatrix} 1 & 0 \\ -\hat{b}_{\mathcal{B}} & -\hat{B} \end{pmatrix} \quad \text{и} \quad (Q^{\mathcal{B}})^{-1}Q^I = \begin{pmatrix} 1 & 0 \\ -\hat{b}_{\mathcal{B}} + \hat{B}p^I & \hat{B}\hat{A}_I^{-1} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -T_{\mathcal{B}*}^I & \end{pmatrix}.$$

Поскольку база \mathcal{B} отличается от базы I только j -ым элементом, то матрица

$$\begin{pmatrix} 1 & 0 \\ -T_{\mathcal{B}*}^I & \end{pmatrix}$$

отличается от единичной матрицы только j -ой строчкой (нумерация строчек начинается с нуля), равной $-t_k^I$. Таким образом, матрица

$$((Q^{\mathcal{B}})^{-1}Q^I)^{-1}$$

отличается от единичной матрицы только j -ой строчкой равной

$$\left(-\frac{t_{k0}^I}{t_{kj}^I}, \dots, -\frac{t_{k,j-1}^I}{t_{kj}^I}, -\frac{1}{t_{kj}^I}, -\frac{t_{k,j+1}^I}{t_{kj}^I}, \dots, -\frac{t_{kn}^I}{t_{kj}^I} \right)$$

Соотношения (45) следуют из равенства

$$T^{\mathcal{B}} = T^I((Q^{\mathcal{B}})^{-1}Q^I)^{-1}$$

и правила перемножения матриц. Отметим, что число элементарных операций, необходимых для пересчета симплекс-таблицы по формулам (45), не превосходит $(2n+1) \times (m+n+1)$.

Дадим теперь пошаговое описание столбцовой формы симплекс-метода.

Алгоритм 2. [Симплекс-метод. Столбцовая форма.]

Шаг 1. Построить симплекс-таблицу T^I , соответствующую начальной допустимой строковой базе I .

Шаг 2. Найти $j \in \{1, 2, \dots, n\}$, для которого $t_{0j}^I < 0$. Если таких j нет, то стоп; текущая база оптимальна.

Шаг 3. Найти индекс k , при котором $t_{kj}^I > 0$ и достигается минимум

$$\frac{t_{k0}^I}{t_{kj}^I} = \min_{l: t_{lj}^I > 0} \frac{t_{l0}^I}{t_{lj}^I}.$$

Если при всех k имеет место неравенство $t_{kj}^I \leq 0$, то стоп; оптимального плана не существует, так как линейная форма не ограничена снизу на множестве допустимых планов.

Шаг 4. Пересчитать симплекс-таблицу по формулам (45), заменить j -ый элемент в базе I на k и вернуться на шаг 2.

Пример 2.21. Решим столбцовым симплекс-методом задачу линейного программирования

$$\begin{cases} \min(-x_1 - x_2) \\ 2x_1 + x_2 \leq 6, \\ x_1 + 2x_2 \leq 6, \\ x_1 \geq 0, \\ x_2 \geq 0. \end{cases}$$

В качестве начальной допустимой базы можно положить $\{3, 4\}$. Ей будет соответствовать таблица

$$\left(\begin{array}{c|cc} 0 & -1 & -1 \\ \hline 6 & \boxed{2} & 1 \\ 6 & 1 & 2 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{array} \right)$$

На втором шаге можно выбрать $j = 1$ или $j = 2$. Положим $j = 1$. На третьем шаге находим $k = 1$. Таким образом, в текущей базе первый элемент заменится на 1, и следующая база равна $\{1, 4\}$. Вычислим симплекс-таблицу для новой базы по формулам (2.5). Для этого из нулевого столбца вычтем первый столбец, умноженный на $6/2 = 3$, из второго столбца

вычтем первый, умноженный на $1/2$ и первый столбец поделим на -2 . В результате получим симплекс-таблицу:

$$\left(\begin{array}{c|cc} 3 & 1/2 & -1/2 \\ \hline 0 & -1 & 0 \\ 3 & -1/2 & \boxed{3/2} \\ 3 & 1/2 & 1/2 \\ 0 & 0 & -1 \end{array} \right)$$

Полученная база не является оптимальной, о чем свидетельствует наличие отрицательного элемента в нулевой строке. На втором и третьем шагах симплекс-метода находим $j = 2$ и $k = 2$. Следовательно, следующая база равна $\{1, 2\}$, а соответствующая ей симплекс-таблица имеет вид:

$$\left(\begin{array}{c|cc} 6 & 1/3 & 1/3 \\ \hline 0 & -1 & 0 \\ 0 & 0 & -1 \\ 2 & 2/3 & -1/3 \\ 2 & -1/3 & 2/3 \end{array} \right)$$

Опорный план, соответствующий этой базе, является решением системы линейных уравнений, которая получается обращением в равенство всех неравенств с номерами из базы. В нашем случае — это первое и второе неравенство. Таким образом, оптимальный план может быть найден как решение системы линейных уравнений

$$\begin{cases} x_1 + 2x_2 = 6, \\ 2x_1 + x_2 = 6. \end{cases}$$

Заметим, что в ряде случаев, в том числе и только что рассмотренном, опорный план можно найти проще. Действительно, нулевой столбец в симплекс-таблице равен $\hat{b} - \hat{A}p^I$, и, следовательно, его компоненты суть величины невязок неравенств на опорном плане. В приведенном примере невязки для оптимального плана у третьего и четвертого неравенств равны 2 и поэтому $x_1 = 2$ и $x_2 = 2$.

Глава 3

Двойственность в линейном программировании

3.1. Теорема двойственности

Основная идея двойственности состоит в совместном изучении пары ЗЛП: исходной и так называемой двойственной к ней.

Определение 3.1. ЗЛП с переменными u_1, \dots, u_m

$$\min(b_1 u_1 + b_2 u_2 + \dots + b_m u_m) \quad (46)$$

$$a_{1j} u_1 + a_{2j} u_2 + \dots + a_{mj} u_m = c_j \quad (j = 1, 2, \dots, n) \quad (47)$$

называется *двойственной* к ЗЛП (5-7). При этом задача (5-7) называется *прямой*.

ЗЛП (46-47) будем записывать также в матричной форме

$$\begin{aligned} \min ub \\ uA = c, \end{aligned}$$

где u — вектор-строка длины m .

Заметим, что переменная u_i двойственной задачи не ограничена по знаку.

Определение 3.2. Переменную u_i задачи (46–47) будем называть соответствующей i -му уравнению прямой задачи.

Лемма 3.3. Пусть $p = (p_1, \dots, p_n)^\top$ — план задачи (5–7), а $t = (t_1, \dots, t_n)$ — план задачи (46–47), тогда

$$c_1 p_1 + c_2 p_2 + \dots + c_n p_n \leq b_1 t_1 + b_2 t_2 + \dots + b_m t_m.$$

ДОКАЗАТЕЛЬСТВО. Имеем

$$\begin{cases} Ap = b, \\ p \geq 0 \end{cases} \quad \text{и} \quad tA \geq c. \quad (48)$$

Так как $p \geq 0$, то из $tA \geq c$ следует $tAp \geq cp$, откуда, ввиду $Ap = b$, получаем $tb \geq cp$. ■

Следствие 3.4. Если p, t — планы прямой и двойственной задачи соответственно, причем значения целевых функций на них совпадают, т.е. $cp = ut$, то p, t — оптимальные планы прямой и двойственной задач соответственно.

Следствие 3.5. Если множество M планов прямой задачи (5–7) не пусто, и целевая функция cx не ограничена сверху на M , то множество N планов двойственной задачи (46–47) пусто.

Следствие 3.6. Если множество N планов двойственной задачи (46–47) не пусто, и целевая функция ub не ограничена снизу на N , то множество M планов прямой задачи (5–7) пусто.

Теорема 3.7. Если ЗЛП (5–7) имеет оптимальный план, то ЗЛП (46–47) также имеет оптимальный план и значения целевых функций на них совпадают.

ДОКАЗАТЕЛЬСТВО. Применим к ЗЛП (5–7) какой-либо вариант симплекс-метода, гарантирующий от заикливания. Через конечное число получаются: база

$$a_{j_1}, \dots, a_{j_r},$$

соответствующая ей симплексная таблица

$$Q = (q_{ij}) \quad (i = 0, 1, \dots, r; j = 0, 1, \dots, n),$$

52 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

такая, что

$$q_{i0} \geq 0 \quad (i = 1, 2, \dots, r), \quad q_{0j} \geq 0 \quad (j = 1, 2, \dots, n),$$

и план

$$p = (p_1, p_2, \dots, p_n)^\top,$$

в котором

$$\begin{aligned} p_{i_j} &= q_{i0}, \\ p_j &= 0, \quad \text{для остальных компонент} \end{aligned}$$

и

$$cp = q_{00}.$$

Так как столбцы a_{j_1}, \dots, a_{j_r} линейно независимы, то в A найдется невырожденная подматрица

$$B = (b_{\mu\nu}), \quad \text{где } b_{\mu\nu} = a_{i_\mu j_\nu} \quad (\mu = 1, 2, \dots, r; \nu = 1, 2, \dots, r).$$

Пусть $v^o = (v_1^o, \dots, v_r^o)$ — решение системы

$$b_{1\nu}v_1 + b_{2\nu}v_2 + \dots + b_{r\nu}v_r = c_{j_\nu} \quad (\nu = 1, 2, \dots, r). \quad (49)$$

Покажем, что вектор $t = (t_1, \dots, t_m)$, где

$$\begin{aligned} t_{i_\mu} &= v_\mu^o \quad (\mu = 1, 2, \dots, r), \\ t_i &= 0, \quad \text{для остальных компонент.} \end{aligned}$$

является оптимальным планом для двойственной задачи.

Действительно,

$$ta_{j_\mu} = c_{j_\mu} \quad (\mu = 1, 2, \dots, r) \quad (50)$$

и

$$ta_j - c_j = t \sum_{i=1}^r q_{ij} a_{j_i} - c_j = \sum_{i=1}^r q_{ij} ta_{j_i} - c_j = \sum_{i=1}^r q_{ij} ca_{j_i} - c_j = q_{0j} \geq 0,$$

т.е. t удовлетворяет (50)

Кроме того,

$$tb = \sum_{i=1}^r q_{i0} ta_{a_{j_i}} = \sum_{i=1}^r q_{i0} c_{j_i} = q_{00}.$$

Для доказательства теоремы осталось воспользоваться Следствием 3.4. ■

Замечание 3.8. Предложенное доказательство дает способ построения оптимального плана t двойственной ЗЛП (46–47) по уже найденной базе $a_{j_1}, a_{j_2}, \dots, a_{j_r}$ оптимального плана прямой ЗЛП (5–7). План t является решением (вообще говоря, неединственным), 50. Для любого такого решения $ta_j - c_j = q_{0j}$ и $tb = q_{00}$.

Пример 3.9. Составим и решим задачу, двойственную к ЗЛП из примера 2.6

По определению двойственная ЗЛП имеет следующий вид:

$$\begin{aligned} & \min(6u_1 + 6u_2) \\ & \begin{cases} 2u_1 + u_2 \geq 1, \\ u_1 + 2u_2 \geq 1, \\ u_1 \geq 0, \\ u_2 \geq 0. \end{cases} \end{aligned}$$

Так как оптимальный план $p^o = (2, 2, 0, 0)^\top$ прямой ЗЛП соответствует базе a_1, a_2 , то система (49) примет вид

$$\begin{cases} 2v_1 + v_2 = 1, \\ v_1 + 2v_2 = 1, \end{cases}$$

Решение этой системы $t = (1/3, 1/3)$ является оптимальным планом двойственной ЗЛП.

Замечание 3.10. Предположим, что целевая функция cx не ограничена на множестве M планов прямой задачи (5–7). Тогда по следствию 3.5 условия двойственной задачи несовместны.

Через конечное число итераций симплекс-метода, гарантирующего от заикливания, мы получим базу $a_{j_1}, a_{j_2}, \dots, a_{j_r}$ и соответствующую ей симплексную таблицу $Q = (q_{ij})$ ($i = 0, 1, \dots, r; j = 0, 1, \dots, n$), для которой найдется l , такое, что $q_{0l} < 0$ и $q_{il} \leq 0$ ($i = 1, 2, \dots, r$).

В данном случае доказательство теоремы 3.7 дает нам способ найти такую линейную с неотрицательными коэффициентами комбинацию неравенств двойственной задачи, которая является неравенством, противоречащим ее l -му неравенству. Действительно, домножив неравенство $ua_{j_i} \geq c_{j_i}$ на неотрицательное число $-q_{il}$ и просуммировав по $i = 1, 2, \dots, r$, получим

$$-u \sum_{i=1}^r q_{il} a_{j_i} \geq - \sum_{i=1}^r q_{il} c_{j_i}.$$

54 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

Так как

$$a_l = \sum_{i=1}^r q_{il} a_{j_i} \quad \text{и} \quad q_{0l} = \sum_{i=1}^r q_{il} c_{j_i} - c_l,$$

то последнее неравенство переписывается как

$$u a_l \leq q_{0l} + c_l.$$

Поскольку $q_{0l} < 0$, то имеем $u a_l < c_l$, что противоречит l -му неравенству в (47).

Замечание 3.11. Как показывает следующий пример, из несовместности условий прямой ЗЛП не следует неограниченность линейной формы в двойственной ЗЛП, более того не следует даже совместность условий в двойственной ЗЛП.

Пример 3.12. Рассмотрим ЗЛП

$$\begin{cases} \max x_1 \\ x_1 - x_2 = 1, \\ x_3 = -1, \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

и двойственную к ней:

$$\begin{cases} \min(u_1 - u_2) \\ u_1 \geq 1, \\ -u_1 \geq 0, \\ u_2 \geq 0. \end{cases}$$

Очевидно, что обе задачи не имеют планов.

Теорема 3.13. Если ЗЛП (46–47) имеет оптимальный план, то ЗЛП (5–7) также имеет оптимальный план и значения целевых функций на них совпадают.

Доказательство. Для доказательства приведем ЗЛП (46–47) к каноническому виду. Согласно разделу 1.3 будем иметь:

$$\begin{cases} \max(-b_1(u'_1 - u''_1) - b_2(u'_2 - u''_2) - \dots - b_m(u'_m - u''_m)) \\ a_{1j}u'_1 + \dots + a_{mj}u'_m - a_{1j}u''_1 - \dots - a_{mj}u''_m - v_j = c_j \quad (j = 1, 2, \dots, n) \\ u'_i \geq 0, \quad u''_i \geq 0 \quad (i = 1, 2, \dots, m) \\ v_j \geq 0 \quad (j = 1, 2, \dots, n) \end{cases} \quad (51)$$

Задача, двойственная к ЗЛП (51), имеет вид:

$$\begin{cases} \min(c_1y_1 + c_2y_2 + \dots + c_ny_n) \\ a_{i1}y_1 + a_{i2}y_2 + \dots + a_{in}y_n \geq -b_i \quad (i = 1, 2, \dots, m) \\ -a_{i1}y_1 - a_{i2}y_2 - \dots - a_{in}y_n \geq b_i \quad (i = 1, 2, \dots, m) \\ -y_j \geq 0 \quad (j = 1, 2, \dots, n) \end{cases} \quad (52)$$

Положим $x_j = -y_j$ ($j = 1, 2, \dots, n$) и для каждого i заменим пару неравенств

$$a_{i1}y_1 + a_{i2}y_2 + \dots + a_{in}y_n \geq -b_i \quad \text{и} \quad -a_{i1}y_1 - a_{i2}y_2 - \dots - a_{in}y_n \geq b_i$$

эквивалентным уравнением

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = -b_i.$$

Получим

$$\begin{cases} \max(c_1x_1 + c_2x_2 + \dots + c_nx_n) \\ a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = -b_i \quad (i = 1, 2, \dots, m) \\ x_j \geq 0 \quad (j = 1, 2, \dots, n) \end{cases}$$

Полученная ЗЛП эквивалента исходной ЗЛП (5-7).

Пусть $u = (u_1, \dots, u_m)$ — оптимальный план ЗЛП (46-47). Положим

$$\begin{aligned} u'_i &= \max\{0, u_i\}, \\ u''_i &= \max\{0, -u_i\} && (i = 1, 2, \dots, m) \\ v_j &= a_{1j}u'_j + \dots + a_{mj}u'_m - a_{1j}u''_j + \dots + a_{mj}u''_m - c_j && (j = 1, 2, \dots, n) \\ t &= (u_1, \dots, u_m, u'_1, \dots, u''_m, v_1, \dots, v_n). \end{aligned}$$

Очевидно, что t — оптимальный план (51). Тогда по теореме 3.7 задача (52), а, следовательно, и (5-7) тоже имеет оптимальный план. ■

Пример 3.14. Рассмотрим ЗЛП

$$\begin{cases} \max(x_1 + x_2) \\ 2x_1 + x_2 = 1, \\ 3x_1 + 2x_2 = 1, \\ x_1 \geq 0, x_2 \geq 0 \end{cases}$$

56 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

и двойственную к ней:

$$\begin{cases} \min(u_1 + u_2) \\ 2u_1 + 3u_2 \geq 1, \\ u_1 + 2u_2 \geq 1. \end{cases}$$

Так как прямая задача не имеет планов (единственное решение $x_1 = 1$, $x_2 = -1$ ее системы уравнений не удовлетворяет условиям неотрицательности), а двойственная имеет (например, $u_1 = -1$, $u_2 = 1$), то из теоремы 3.13 следует, что значение линейной функции на множестве планов двойственной задачи не ограничено.

Замечание 3.15. При доказательстве теоремы 3.13 по существу показано, что ЗЛП (5–7) является двойственной к ЗЛП (46–47) и, таким образом, можно говорить о паре двойственных задач линейного программирования. Аналогично, воспользовавшись результатами раздела 1.3, можно написать двойственную задачу к любой ЗЛП, предварительно приведя последнюю к каноническому виду.

Упражнение 3.16. Доказать, что следующая пара задач является парой двойственных ЗЛП:

$$\begin{cases} \min(c_1x_1 + c_2x_2 + \dots + c_nx_n) \\ \begin{cases} a_{i1}x_1 + a_{i2}x_2 + \dots + a_{im}x_m = b_i & (i = 1, 2, \dots, m), \\ x_j \geq 0 & (j = 1, 2, \dots, n) \end{cases} \end{cases} \quad (53)$$

и

$$\begin{cases} \min(b_1u_1 + b_2u_2 + \dots + b_mu_m) \\ \begin{cases} a_{1j}u_1 + a_{2j}u_2 + \dots + a_{mj}u_m = c_j & (j = 1, 2, \dots, n), \\ u_i \geq 0 & (i = 1, 2, \dots, m). \end{cases} \end{cases} \quad (54)$$

Определение 3.17. Пара задач (53, 54) называется парой двойственных задач в симметричной форме.

Полученные результаты о паре двойственных ЗЛП (5–7) и (46–47) можно переформулировать в виде следующей теоремы

Теорема 3.18 (Теорема двойственности). Пусть дана пара двойственных ЗЛП (5–7) и (46–47), тогда справедливо одно и только одно из следующих утверждений:

3.2. ДОПОЛНЯЮЩАЯ НЕЖЕСТКОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ 57

а) обе задачи имеют оптимальные планы с равными значениями целевых функций на них;

б) одна из задач имеет планы, но не имеет оптимального плана (целевая функция не ограничена на множестве планов), а условия другой задачи не совместны;

в) условия обеих задач не совместны.

Упражнение 3.19. Сформулируйте и докажите теорему двойственности для пары двойственных задач (53, 54).

3.2. Дополняющая жесткость в линейном программировании

Докажем важное свойство оптимальных планов пары двойственных задач, для которых реализуется случай а) теоремы двойственности.

Теорема 3.20. Пусть $p = (p_1, \dots, p_n)^T$ и t — планы задач (5–7) и (46–47) соответственно. Тогда условие

$$(tA - c)p = 0 \quad (55)$$

является необходимым и достаточным для оптимальности каждого из планов p и t .

Доказательство. Подправим рассуждения из доказательства леммы 3.3. Для планов p и t выполняются соотношения (48), поэтому

$$(tA - c)p = 0 \quad \Leftrightarrow \quad tAp = cp \quad \Leftrightarrow \quad tb = cp$$

Последнее условие является необходимым и достаточным для оптимальности планов. ■

Замечание 3.21. Условие (55) означает, что по крайней мере одно из неравенства

$$x_0 \geq 0 \quad \text{и} \quad a_{1j}u_1 + \dots + a_{1m}u_m \geq c_j$$

на оптимальных планах должно выполняться как равенство.

Это условие может быть использовано для доказательства оптимальности плана и для построения оптимального плана прямой задачи по

58 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

оптимальному плану двойственной. Например, если оптимальный план прямой задачи невырожден, т.е.

$$p_{j_i} > 0 \quad (i = 1, 2, \dots, r),$$

то из (55) следует, что для оптимального плана t двойственной задачи (46–47) должно выполняться равенство

$$ta_{j_i} = c_{j_i} \quad (i = 1, 2, \dots, r).$$

Заметим, что полученные условия совпадают с системой (50), появившейся при доказательстве теоремы 3.7.

Упражнение 3.22. Сформулируйте и докажите аналог теоремы 3.20 для пары двойственных задач (53, 54).

Определение 3.23. Ограничение-неравенство называется *жестким* в ЗЛП, если для всякого оптимального плана этой ЗЛП оно выполняется как равенство, и *нежестким* в противном случае.

Можно доказать следующее усиление теоремы 3.20.

Теорема 3.24 (Свойство дополняющей нежесткости). Если задача (5–7) и задача (46–47) имеют оптимальные планы, то в каждой паре ограничений

$$x_j \geq 0, \quad ua_j \geq c_j$$

одно неравенство жесткое, а одно нежесткое.

Пример 3.25. Докажем, что план $t = (7, 0, 1)^T$ является оптимальным для задачи

$$\begin{cases} \min(-3u_1 - 5u_2 + 11u_3) \\ -3u_1 - 6u_2 + 11u_3 \geq -10, \\ -u_1 - 2u_2 + 4u_3 \geq -3, \\ -u_2 + 2u_3 \geq -1, \\ u_i \geq 0 \quad (i = 1, 2, 3). \end{cases}$$

Эту задачу можно рассмотреть как двойственную к задаче

$$\begin{cases} \max(-10x_1 - 3x_2 - x_3) \\ -3x_1 - x_2 + x_4 = -3, \\ -6x_1 - 2x_2 - x_3 + x_5 = -5, \\ 11x_1 + 4x_2 + 2x_3 + x_6 = 11, \\ x_j \geq 0 \quad (j = 1, 2, \dots, 6). \end{cases}$$

Подставим t в ограничения первой задачи. Неравенства

$$\begin{aligned} -u_2 + 2u_3 &\geq -1, \\ u_1 &\geq 0, \\ u_3 &\geq 0 \end{aligned}$$

выполнятся как строгие. Следовательно, компоненты p_3, p_4, p_6 плана p должны быть нулевыми. Остальные компоненты p найдем, подставив p в ограничения-равенства второй задачи:

$$\begin{cases} -3p_1 - p_2 = -3, \\ -6p_1 - 2p_2 + p_5 = -5, \\ 11p_1 + 4p_2 = 11, \end{cases}$$

откуда $p = (1, 0, 0, 0, 1, 0)^\top$. По теореме 3.20 планы t и p — оптимальные.

3.3. Задачи

3.1. Проверить оптимальность плана следующих задач:

$$p = (0, 1, 1),$$

$$\max(-4x_1 + x_2 + 5x_3)$$

$$\begin{cases} x_2 + x_3 = 2, \\ 3x_1 + 2x_2 - x_3 = 1, \\ x_j \geq 0 \quad (j = 1, 2, 3); \end{cases}$$

$$p = (3, 4, 0),$$

$$\max(x_1 + 2x_2 - 5x_3)$$

$$\begin{cases} x_1 - 3x_2 + 11x_3 = -9, \\ 3x_1 - x_2 + 9x_3 = 5, \\ x_j \geq 0 \quad (j = 1, 2, 3); \end{cases}$$

$$p = (1, 1, 1, 1, 1, 1, 1),$$

$$\max(5x_1 + 4x_2 + 3x_3 + 11x_4 + 10x_5 - 2x_6 - 3x_7 - 6x_8)$$

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 - x_6 + x_7 - x_8 = 14, \\ 2x_1 + 3x_2 + x_3 + 5x_4 - 6x_5 - x_6 - x_7 - 2x_8 = 13, \\ 3x_1 + x_2 + 2x_3 + 6x_4 + 4x_5 - x_6 - 2x_7 - 4x_8 = 9. \end{cases}$$

60 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

3.2. Для каждого неравенства ЗЛП

$$\begin{aligned} & \max(10x_1 + 3x_2 + x_3) \\ & \begin{cases} 3x_1 + x_2 \geq 3, \\ 6x_1 + 2x_2 + x_3 \geq 5, \\ -11x_1 - 4x_2 - 2x_3 \geq -11, \\ x_j \geq 0 \quad (j = 1, 2, 3) \end{cases} \end{aligned}$$

определить, является ли оно жестким.

3.4. Двойственный симплекс-метод

Определение 3.26. Пусть B — база, такая, что $q_{0j} \geq 0$. Тогда B называется *двойственно допустимой базой*, или *псевдобазой*, таблица Q — *двойственно-допустимой таблицей*, а план p , такой, что $p_j = 0$ ($j \in \mathcal{N}$), $p_{s_i} = q_{i0}$ ($i = 1, 2, \dots, r$), — *псевдобазой*.

Идея двойственного симплекс-метода состоит в том, чтобы переходя от псевдоплана к псевдоплану получить такой псевдоплан, для которого $q_{i0} \geq 0$ ($i = 1, 2, \dots, s$).

Алгоритм 3. [Двойственный симплекс-метод]

Шаг 1. Найти матрицу Q , соответствующую псевдобазе $\{a_j : j \in \mathcal{B}\}$, и вычислить q_{00} и q_{0j} .

Шаг 2. Проверить, выполняется ли для всех $i \in \{1, 2, \dots, r\}$ неравенство $q_{i0} \geq 0$. Если да, то стоп (p — оптимальный план). Если нет, то выбрать k так, чтобы $q_{k0} < 0$.

Шаг 3. Проверить, выполняется ли для всех $j \in \{1, 2, \dots, n\}$ неравенство $q_{kj} \geq 0$. Если да, то стоп (условия задачи несовместны). Если нет, то найти такое l , что

$$\frac{q_{0l}}{-q_{kl}} = \min_{j: q_{kl} < 0} \frac{q_{0j}}{-q_{kj}} \quad (56)$$

Шаг 4. В множестве \mathcal{B} заменить s_k на l и перейти к шагу 1.

Определение 3.27. Строку k , столбец l и элемент q_{kl} в алгоритме 3 будем называть *направляющими*.

Нетрудно проверить, что выбор направляющего столбца l по формуле (56) гарантирует сохранение двойственной допустимости и уменьшение значения линейной формы при $q_{0j} > 0$ ($j \in \mathcal{N}$).

Пример 3.28. Решим двойственным симплекс-методом ЗЛП

$$\begin{cases} \max(-x_1 - x_2 - x_3 - x_4) \\ -x_1 + x_2 - 2x_3 + 2x_4 + x_5 = -1, \\ x_1 - 2x_2 + 3x_3 - 4x_4 + x_6 = -2, \\ x_j \geq 0 \quad (j = 1, 2, \dots, 6). \end{cases}$$

Выбрав за базис a_5 и a_6 , получим исходную двойственно-допустимую таблицу

$$\left(\begin{array}{c|cccccc} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ \hline -1 & -1 & 1 & -2 & 2 & 1 & 0 \\ -2 & 1 & 2 & 3 & \boxed{-4} & 0 & 1 \end{array} \right).$$

Выбирая k , соответствующее наименьшему из q_{i0} , получим $k = 2$, и далее по (56) имеем $l = 4$:

$$\left(\begin{array}{c|cccccc} -1/2 & 5/4 & 1/2 & 7/4 & 0 & 0 & 1/4 \\ \hline -2 & \boxed{-1/2} & 0 & -1/2 & 0 & 1 & 1/2 \\ -1/2 & -1/4 & 1/2 & -3/4 & 1 & 0 & -1/4 \end{array} \right).$$

Теперь $k = 2$, $l = 2$:

$$\left(\begin{array}{c|cccccc} -11/2 & 0 & 1/2 & 1/2 & 0 & 5/2 & 3/2 \\ \hline 4 & 1 & 0 & 1 & 0 & -2 & -1 \\ 3/2 & 0 & 1/2 & -1/2 & 1 & -1/2 & -1/2 \end{array} \right).$$

Последняя таблица соответствует оптимальному плану $(4, 0, 0, 3/2, 0, 0)$.

Замечание 3.29. Если $a_{s_1}, a_{s_2}, \dots, a_{s_r}$ — псевдобаза ЗЛП (5-7), то любое решение t системы

$$ua_{s_i} = c_{s_i} \quad (i = 1, 2, \dots, r)$$

62 ГЛАВА 3. ДВОЙСТВЕННОСТЬ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

будет планом двойственной задачи (46–47), так как

$$ta_j = \sum_{i=1}^r q_{ij} ta_{s_i} = \sum_{i=1}^r q_{ij} c_{s_i} \leq c_j,$$

из-за $q_{0j} \geq 0$.

Таким образом, движению по псевдопланам прямой задачи в двойственном симплекс-методе соответствует движению по планам двойственной задачи. Двойственный симплекс-метод удобно применять, если решается последовательность задач, различающихся только вектором b , так как в этом случае оптимальный план предыдущей задачи будет двойственно допустимым планом последующей задачи.

3.5. Задачи

3.3. Найти решение задачи линейного программирования

$$\begin{aligned} & \max(3x_1 + 5x_2 + 4x_3) \\ & \begin{cases} x_1 + 2x_2 + 3x_3 \geq 1, \\ 2x_1 + 3x_2 + x_3 \geq 1, \\ 3x_1 + x_2 + 2x_3 \geq 1, \\ 6x_1 + 6x_2 + 6x_3 \geq 1, \\ 2x_1 + \quad \quad + 4x_3 \geq 1 \end{cases} \end{aligned} \quad (57)$$

сначала двойственным симплекс-методом, а затем прямым методом, применив его к двойственной задаче.

Глава 4

Целочисленное линейное программирование

4.1. Идея правильных отсечений

Определение 4.1. Задачу линейного программирования, в которой на значения всех или части переменных наложено требование целочисленности, называют *задачей целочисленного линейного программирования* (ЗЦЛП). В первом случае говорят о задаче *полностью целочисленного линейного программирования*, а во втором о задаче *частично целочисленного линейного программирования*. Если коэффициенты ограничений и функции цели ЗЦЛП целочисленны, то говорят о *целочисленной* ЗЦЛП.

Пусть p — оптимальный план ЗЛП, полученной из ЗЦЛП отбрасыванием требования целочисленности. Очевидно, что значение максимизируемой функции на любом плане ЗЦЛП не превосходит значения этой функции на плане p . Следовательно, если p удовлетворяет условия целочисленности, то он оптимальный и для ЗЦЛП. Однако так бывает далеко не всегда. Округление же компонент до ближайшего целого может не только не дать оптимального плана ЗЦЛП, но и вывести за пределы допустимых решений ЗЛП. Американские ученые Дж. Данциг, Д. Фалкерсон и С. Джонсон предложили в этом случае добавлять к условиям ЗЛП такое линейное неравенство (так называемое *правильное отсечение*), которому любой план ЗЦЛП удовлетворяет, а p не удовлетворяет. После этого

64 ГЛАВА 4. ЦЕЛОЧИСЛЕННОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

решается новая ЗЛП, строится новое отсечение и т.д., пока либо не получится оптимальный план, удовлетворяющий условиям целочисленности, либо не будет выявлена несовместность условий полученной ЗЛП.

Способ построения правильных отсечений, гарантирующий конечность описанной процедуры, был опубликован американским математиком Р.Гомори в 1958 г.

4.2. Постановка задачи

Так как оптимальный план ЗЛП после правильного отсечения не является планом новой ЗЛП, но является ее двойственным планом, то для решения новой ЗЛП выгоднее использовать двойственный симплекс-метод (или применять прямой симплекс-метод для двойственной задачи). Кроме того, так как правильное отсечение является неравенством, то удобнее использовать столбцовую форму записи и считать, что ограничения исходной задачи заданы в форме неравенств.

Пусть $A = (a_{ij})$ — целочисленная матрица размеров $m \times n$, a_1, a_2, \dots, a_n — ее столбцы, $b = (b_1, b_2, \dots, b_m)^\top$, $c = (c_1, c_2, \dots, c_n)$ — целочисленные векторы. Рассмотрим ЗЦЛП относительно вектора переменных $u = (u_1, u_2, \dots, u_m)$:

$$\begin{aligned} & \max ub \\ & \begin{cases} ua_j \leq c_j & (j = 1, 2, \dots, n), \\ u_i \in \mathbf{Z} & (i = 1, 2, \dots, m). \end{cases} \end{aligned}$$

Запишем также эту задачу в матричной форме:

$$\begin{aligned} & \max ub \\ & \begin{cases} uA \leq c, \\ u \in \mathbf{Z}^m. \end{cases} \end{aligned}$$

4.3. Циклический алгоритм Гомори

Лемма 4.2 (Гомори). Пусть

$$u = a_0 + \sum_{i=1}^m a_i(-u_i), \quad (58)$$

причем $u \in \mathbf{Z}$, $u_i \in \mathbf{Z}$, $u_i \geq 0$ ($i = 1, 2, \dots, m$), $a_0 \notin \mathbf{Z}$. Тогда

$$-\{a_0\} - \sum_{i=1}^m \{a_i\}(-u_i) \geq 0. \quad (59)$$

ДОКАЗАТЕЛЬСТВО. Так $u \in \mathbf{Z}$, то из (58) следует, что

$$\{a_0\} - \sum_{i=1}^m \{a_i\} u_i \in \mathbf{Z}. \quad (60)$$

Так как $\{a_0\} < 1$, то из (60) получаем

$$\sum_{i=1}^m \{a_i\} u_i \geq \{a_0\},$$

откуда и следует (59). ■

Определение 4.3. Неравенство (59) называется *отсечением Гомори*.

Чтобы использовать отсечение Гомори при решении ЗЦЛП удобно ввести новую неотрицательную переменную

$$u = -\{a_0\} - \sum_{i=1}^m \{a_i\}(-u_i). \quad (61)$$

Очевидно, что условия $u \geq 0$ и (61) эквивалентны неравенству (59).

Определение 4.4. Циклический алгоритм, в котором на каждом шаге строится отсечение Гомори, называется *циклическим алгоритмом Гомори*.

Пример 4.5. Решим циклическим алгоритмом Гомори ЗЦЛП:

$$\begin{array}{l} \max(u_1 + u_2) \\ \left\{ \begin{array}{ll} u_1 & \geq 0, \\ u_2 & \geq 0, \\ -2u_1 + 3u_2 & \leq 4, \\ 3u_1 - 2u_2 & \leq 1, \\ u_1 & \in \mathbf{Z}, \\ u_2 & \in \mathbf{Z}. \end{array} \right. \end{array} \quad (62)$$

66 ГЛАВА 4. ЦЕЛОЧИСЛЕННОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

Запишем ограничения задачи в виде следующей таблицы, к которой применим прямой симплекс-метод:

$$\left(\begin{array}{c|cc} 0 & -1 & -1 \\ \hline 0 & -1 & 0 \\ 0 & 0 & -1 \\ 4 & -2 & 3 \\ 1 & \boxed{3} & -2 \end{array} \right) \left(\begin{array}{c|cc} 1/3 & 1/3 & -5/3 \\ \hline 1/3 & 1/3 & -2/3 \\ 0 & 0 & -1 \\ 14/3 & 2/3 & \boxed{-5/3} \\ 0 & -1 & 0 \end{array} \right) \left(\begin{array}{c|cc} 5 & 1 & 1 \\ \hline 11/5 & 3/5 & 2/5 \\ 14/5 & 2/5 & 3/5 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{array} \right)$$

Из первой строки получаем

$$u_1 = \frac{11}{5} + \frac{3}{5}(-u_4) + \frac{2}{5}(-u_3).$$

По формуле (59) строим отсечение Гомори:

$$u_5 = -\frac{1}{5} + \frac{3}{5}u_4 + \frac{2}{5}u_3.$$

Припишем коэффициенты этого равенства снизу к таблице и применим симплекс-метод:

$$\left(\begin{array}{c|cc} 5 & 1 & 1 \\ \hline 11/5 & 3/5 & 2/5 \\ 14/5 & 2/5 & 3/5 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \\ \hline -1/5 & \boxed{-3/5} & -2/5 \end{array} \right) \left(\begin{array}{c|cc} 14/3 & 5/3 & 1/3 \\ \hline 2 & 1 & 0 \\ 8/3 & 2/3 & 1/3 \\ 0 & 0 & -1 \\ 1/3 & -5/3 & 2/3 \\ 0 & -1 & 0 \end{array} \right)$$

Из второй строки получаем

$$u_2 = \frac{8}{3} + \frac{2}{3}(-u_5) + \frac{1}{3}(-u_3).$$

По формуле (59) строим отсечение Гомори:

$$u_6 = -\frac{2}{3} + \frac{2}{3}u_5 + \frac{1}{3}u_3.$$

Припишем коэффициенты этого равенства снизу к таблице и применим симплекс-метод:

$$\left(\begin{array}{c|cc} 14/3 & 5/3 & 1/3 \\ \hline 2 & 1 & 0 \\ 8/3 & 2/3 & 1/3 \\ 0 & 0 & -1 \\ 1/3 & -5/3 & 2/3 \\ 0 & -1 & 0 \\ \hline -2/3 & -2/3 & -2/3 \end{array} \right) \left(\begin{array}{c|cc} 4 & 1 & 1 \\ \hline 2 & 1 & 0 \\ 2 & 0 & 1 \\ 2 & 2 & -1 \\ -1 & -3 & 2 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{array} \right) \left(\begin{array}{c|cc} 11/3 & 1/3 & 5/3 \\ \hline 5/3 & 1/3 & 2/3 \\ 2 & 0 & 1 \\ 4/3 & 2/3 & 1/3 \\ 0 & -1 & 0 \\ 1/3 & -1/3 & -2/3 \\ 0 & 0 & -1 \end{array} \right)$$

После введения еще одного отсечения

$$u_7 = -\frac{2}{3} + \frac{1}{3}u_4 + \frac{2}{3}u_6$$

68 ГЛАВА 4. ЦЕЛОЧИСЛЕННОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

и выполнения одного шага симплекс-метода процесс закончится:

$$\left(\begin{array}{c|cc} 11/3 & 1/3 & 5/3 \\ \hline 5/3 & 1/3 & 2/3 \\ 2 & 0 & 1 \\ 4/3 & 2/3 & 1/3 \\ 0 & -1 & 0 \\ 1/3 & -1/3 & -2/3 \\ 0 & 0 & -1 \\ \hline -2/3 & \boxed{-1/3} & -2/3 \end{array} \right) \quad \left(\begin{array}{c|cc} 3 & 1 & 1 \\ \hline 1 & 1 & 0 \\ 2 & 0 & 1 \\ 0 & 2 & -1 \\ 2 & -3 & 2 \\ 1 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{array} \right)$$

Искомая оптимальная точка: $u = (1, 2)$.

4.4. Полностью целочисленный алгоритм

Пусть задача (??) обладает следующим свойством: существует такая псевдобаза

$$F = (a_{s_1}, \dots, a_{s_m}),$$

что $|\det F| = 1$. Тогда для соответствующей симплексной таблицы $Q = (q_{ij})$ выполняется

$$q_{j0} \in \mathbf{Z} \quad (j = 0, 1, \dots, n), \quad q_{0i} \geq 0 \quad (i = 1, 2, \dots, m).$$

Если при этом $q_{j0} \geq 0$ ($j = 1, 2, \dots, n$), то Q — оптимальная симплексная таблица.

Пусть найдется $l \in \{1, 2, \dots, n\}$ такое, что $q_{l0} < 0$. Если при этом

$$q_{li} \geq 0 \quad (i = 1, 2, \dots, m),$$

то множество планов ЗЦЛП пусто. Пусть

$$I = \{i : q_{li} < 0\} \neq \emptyset, \quad \frac{q_{lk}}{|q_{lk}|} = \operatorname{lexmin} \left\{ \frac{q_{li}}{|q_{li}|} : i \in I \right\}.$$

Если направляющий элемент $q_{lk} = -1$, то, сделав шаг симплекс-метода, снова попадем в исходную ситуацию.

Идея полностью целочисленного алгоритма состоит в формировании такого отсечения, чтобы после шага симплекс-метода повторилась исходная ситуация, т.е. направляющий элемент был бы равен -1 .

Лемма 4.6 (Гомори). Пусть

$$u = a_0 + \sum_{i=1}^m a_i(-u_i), \quad (63)$$

причем $u \in \mathbf{Z}$, $u \geq 0$, $u_i \in \mathbf{Z}$, $u_i \geq 0$, $a_0 \in \mathbf{Z}$, $a_i \in \mathbf{Z}$ ($i = 1, 2, \dots, m$). Тогда для любого λ , такого, что $\lambda \geq 1$, $\lambda \in \mathbf{Z}$, справедливо неравенство

$$\left[\frac{a_0}{\lambda} \right] + \sum_{i=1}^m \left[\frac{a_i}{\lambda} \right] (-u_i) \geq 0.$$

Доказательство. Из (63) следует, что

$$\sum_{i=1}^m a_i u_i \leq a_0,$$

поэтому

$$\sum_{i=1}^m \frac{a_i}{\lambda} u_i \leq \frac{a_0}{\lambda}.$$

Вычитая из последнего соотношения неравенства

$$\left\{ \frac{a_i}{\lambda} \right\} u_i \geq 0 \quad (i = 1, 2, \dots, m),$$

получим

$$\sum_{i=1}^m \left(\frac{a_i}{\lambda} - \left\{ \frac{a_i}{\lambda} \right\} \right) u_i \leq \frac{a_0}{\lambda},$$

т.е.

$$\sum_{i=1}^m \left[\frac{a_i}{\lambda} \right] u_i \leq \frac{a_0}{\lambda}.$$

Левая часть полученного неравенства целочисленна, поэтому не превосходит $\left[\frac{a_0}{\lambda} \right]$, что и требовалось. \blacksquare

70 ГЛАВА 4. ЦЕЛОЧИСЛЕННОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

Положим

$$\lambda = \max \{q_{li} : i \in I\}, \quad q_{n+1,i} = \left\lfloor \frac{q_{li}}{\lambda} \right\rfloor \quad (i = 0, 1, \dots, m),$$

$$u_{n+1} = q_{n+1,0} - \sum_{i=1}^m q_{n+1,i} u_{s_i}. \quad (64)$$

По лемме 4.6 неравенство $u_{n+1} \geq 0$ является правильным отсечением. Припишем коэффициенты неравенства в конец таблицы.

Так как $q_{l0} < 0$, то $q_{n+1,0} < 0$ и, следовательно, $(n+1)$ -ую строку можно взять за направляющую в симплекс-методе. Пусть

$$\frac{q_t}{|q_{n+1,t}|} = \text{lexmin} \left\{ \frac{q_i}{|q_{n+1,i}|} : i \in I \right\},$$

т.е. направляющим элементом является $q_{n+1,t}$. Очевидно, что $q_{n+1,t} = -1$. Так как $q_{n+1,i} = -1$ при $i \in I$, то

$$q_t = \text{lexmin} \{q_i : i \in I\}.$$

Доказательство сходимости и описание более тонкого выбора λ можно посмотреть в [3].

Пример 4.7. Решим полностью целочисленным алгоритмом задачу

$$\begin{cases} \min(2u_1 + 5u_2) \\ u_1 \geq 0, \\ u_2 \geq 0, \\ 5u_1 + 12u_2 \geq 4, \\ u_1 \in \mathbf{Z}, \\ u_2 \in \mathbf{Z}. \end{cases}$$

Запишем ограничения задачи в виде следующей таблицы, к которой применим прямой симплекс-метод:

$$\left(\begin{array}{c|cc} 0 & 2 & 5 \\ \hline 0 & -1 & 0 \\ 0 & 0 & -1 \\ -4 & -5 & -12 \end{array} \right)$$

Рассмотрим неравенство, связанное с третьей строкой симплексной таблицы:

$$u_3 = -4 - 5(-u_1) - 12(-u_2) \geq 0.$$

Выберем $\lambda = 12$ и по формуле (64) построим отсечение

$$u_4 = -1 + u_1 + u_2 \geq 0.$$

Его коэффициенты припишем в конец таблицы и выполним один шаг симплекс-метода:

$$\left(\begin{array}{c|cc} 0 & 2 & 5 \\ \hline 0 & -1 & 0 \\ 0 & 0 & -1 \\ -4 & -5 & -12 \\ \hline -1 & \boxed{-1} & -1 \end{array} \right) \quad \left(\begin{array}{c|cc} -2 & 2 & 3 \\ \hline 1 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & -5 & -7 \\ 0 & -1 & -0 \end{array} \right)$$

Сразу получаем оптимальную таблицу и оптимальный целочисленный план $p = (1, 0)$.

4.4.1. Прямой метод целочисленного программирования

В заключение опишем идею так называемого прямого метода ЦЛП.

Пусть $F = (a_{s_1}, \dots, a_{s_m})$ — допустимая унимодулярная база задачи (??)¹ и $Q = (q_{ij})$ — соответствующая симплексная таблица.

Если $q_{0i} \geq 0$ ($i = 1, 2, \dots, m$), то задача (??) решена. В противном случае найдется такое k , что $q_{0k} < 0$. Если при этом $q_{jk} \leq 0$ ($j = 1, 2, \dots, n$), то целевая функция не ограничена сверху на множестве планов задачи (??). В противном случае найдем направляющий элемент q_{lk} прямым симплекс-методом. Если $q_{lk} = 1$, то после шага симплекс-метода ситуация повторяется. Если $q_{lk} \geq 2$, то положим

$$\lambda = q_{lk}, \quad q_{n+1,i} = \left[\frac{q_{li}}{\lambda} \right] \quad (i = 0, 1, \dots, m)$$

¹Для этого достаточно, например, выполнения условий $u \geq 0, c \geq 0$.

72 ГЛАВА 4. ЦЕЛОЧИСЛЕННОЕ ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

и добавим отсечение

$$u_{n+1} = q_{n+1,0} + \sum_{i=1}^m q_{n+1,i}(-u_{s_i}).$$

Заметим, что существуют задачи, в которых описанная процедура приводит к бесконечному процессу. Для получения конечной модификации производящую строку выбирают более сложным образом (см. [3]).

Пример 4.8. Решение ЗЦЛП из примера 4.5 прямым методом приводит к последовательности таблиц:

$$\begin{pmatrix} 0 & -1 & -1 \\ \hline 0 & -1 & 0 \\ 0 & 0 & -1 \\ 4 & -2 & 3 \\ 1 & \boxed{3} & -2 \\ \hline 0 & \boxed{1} & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 & -2 \\ \hline 0 & 1 & -1 \\ 0 & 0 & -1 \\ 4 & 2 & 1 \\ 1 & -3 & \boxed{1} \\ \hline 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 2 & -5 & 2 \\ \hline 1 & -2 & 1 \\ 1 & -3 & 1 \\ 3 & \boxed{5} & -1 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \\ \hline 0 & \boxed{1} & -1 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 5 & -3 \\ \hline 1 & 2 & -1 \\ 1 & 3 & -2 \\ 3 & -5 & \boxed{4} \\ 0 & 0 & -1 \\ 0 & 1 & -1 \\ 0 & -1 & 0 \\ \hline 0 & -2 & \boxed{1} \end{pmatrix} \begin{pmatrix} 2 & -1 & 3 \\ \hline 1 & 0 & 1 \\ 1 & -1 & 2 \\ 3 & \boxed{3} & -4 \\ 0 & -2 & 1 \\ 0 & -1 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ \hline 1 & \boxed{1} & -2 \end{pmatrix} \begin{pmatrix} 3 & 1 & 1 \\ \hline 1 & 0 & 1 \\ 2 & 1 & 0 \\ 0 & -3 & 2 \\ 2 & 2 & -3 \\ 1 & 1 & -1 \\ 1 & 1 & -2 \\ 0 & 0 & -1 \\ \hline 0 & -1 & 0 \end{pmatrix}$$

4.5. Задачи

4.1. Показать, что округление до ближайшего целого компонент оптимального плана ЗЛП

$$\begin{aligned} & \max(8x_1 + 5x_2 + x_3) \\ & \begin{cases} 3x_1 + 2x_2 + x_3 \leq 13, \\ x_j \geq 0 \quad (j = 1, 2, 3); \end{cases} \end{aligned}$$

не приводит к оптимальному плану соответствующей ЗЦЛП.

4.2. Пусть

$$A(\alpha) = \begin{pmatrix} \alpha & -\alpha - 1 \\ -2\alpha + 2 & 2\alpha \end{pmatrix}, \quad c = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Показать, что для каждого ρ найдется такое α , что расстояние от точки $A(\alpha)^{-1}c$ до любой точки $v \in \mathbf{Z}^2$, удовлетворяющей системе будет не меньше ρ .

4.3. Найти x_1, x_2 , чтобы $2x_1 + 5x_2 \leq 3$, $x_1 - 2x_2 \leq 4$, $x_1 \in \mathbf{Z}$, $x_2 \in \mathbf{Z}$, $x_2 > 1000$.

4.4. Решить ЗЦЛП

$$\begin{aligned} & \max(3u_1 + 6u_2 + 11u_3) \\ & \begin{cases} 6u_1 + 10u_2 + 15u_3 \leq 50, \\ u_i \in \mathbf{Z}, \quad u_i \geq 0 \quad (i = 1, 2, 3); \end{cases} \end{aligned}$$

не приводит к оптимальному плану соответствующей ЗЦЛП.

Программа курса

1 Линейное программирование (12 ч.)

1.1. Примеры реальных задач линейного и выпуклого программирования. Задача о смесях, диете, о разрезании (комплекты). Задачи транспортного типа (двухиндексные и многоиндексные). Транспортные задачи с фиксированными доплатами. Примеры задач с разрывами. (2 ч.)

1.2. Приведение ЗЛП к каноническому виду. Симметричная форма. Геометрическая интерпретация. Длина записи входной информации. (1 ч.)

1.3. Симплекс-метод. Прямые алгоритмы, строчечная и столбцовая формы записи, примеры заикливания, способы защиты от заикливания, оценка числа итераций, метод искусственного базиса. (4 ч.)

1.4. Теорема двойственности. Дополняющая нежесткость. Разрыв двойственности в ЦЛП. (3 ч.)

1.5. Двойственные алгоритмы. Геометрическая интерпретация-II. (2 ч.)

2 Системы линейных неравенств (6 ч.)

2.1. Выпуклые множества и конусы. Конечно порожденные и конечно определенные конусы, политопы и полиэдры. Крайние точки и экстремальные лучи. Сопряженность полиэдральных конусов и полярность политопов. (2 ч.)

2.2. Алгоритмы построения остова сопряженного конуса и их применение. Переборный алгоритм. Алгоритм Фурье-Моцкина. (2 ч.)

2.3. Метод эллипсоидов. Полиномиальный алгоритм нахождения рационального решения системы линейных неравенств. (2 ч.)

3 Методы решения задач ЦЛП (4 ч.)

3.1. Идея методов отсечений, геометрическая интерпретация. Выбор вычислительной схемы. (2 ч.)

3.2. Методы отсечений Гомори. (2 ч.)

Литература

1. Данциг Дж. Линейное программирование. Его применения и обобщения – М.: Прогресс, 1966.
2. Схрейвер А. Теория линейного и целочисленного программирования. – М.: Мир, 1991.
3. Ху Т. Целочисленное программирование и потоки в сетях. – М.: Мир, 1974.
4. Шевченко В.Н. Линейное и целочисленное линейное программирование / Уч. пособие. – Горький, 1976.
5. Шевченко В.Н. Линейное программирование и теория линейных неравенств / Уч. пособие. – Горький, 1977.
6. Гасс С. Линейное программирование (методы и приложения). – М.: Физматгиз, 1961.
7. Шевченко В.Н. Качественные вопросы целочисленного линейного программирования. – М.: Физматлит, 1995.
8. Шевченко В.Н. Множества целочисленных решений квадратной системы линейных неравенств / Уч. пособие. – Горький, 1983.